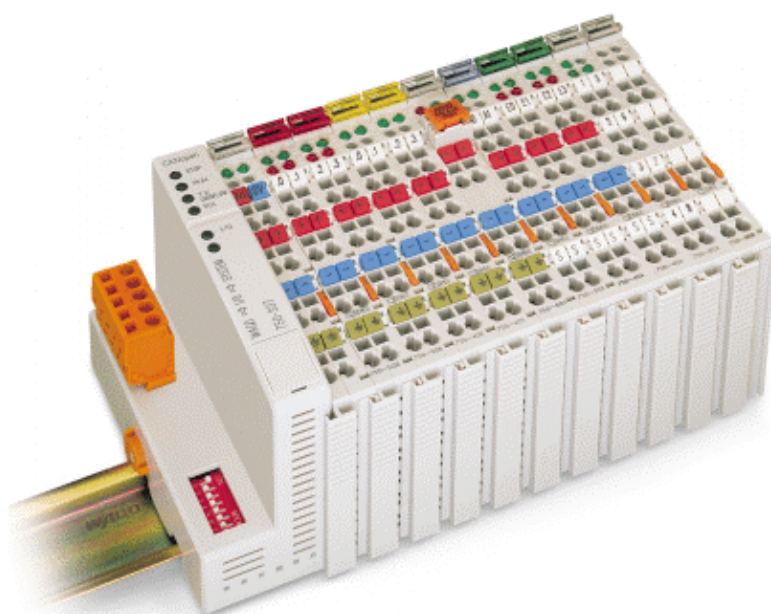


WAGO  SYSTEM **750**

モジュール式 I/O システム

CANopen

750-837、750-838



取扱説明書

技術説明、インストールおよびコンフィグレーション

Ver. 1.0.2 (2010.04.16 日本語版)

Copyright © 2007 by WAGO Kontakttechnik GmbH

All rights reserved.

ワゴコンタクトテクニク社（ドイツ）

Hansastraße 27

D-32423 Minden

電話： +49 (0) 571/8 87-0

ファックス： +49 (0) 571/8 87-1 69

電子メール： info@wago.com

Web： <http://www.wago.com>

技術サポート

電話： +49 (0) 571/8 87-5 55

ファックス： +49 (0) 571/8 87-4 30

電子メール： support@wago.com

本書の作成には万全を期しておりますが、お気づきの点やご意見がございましたら下記までお知らせください。

日本国内問合せ先

ワゴジャパン株式会社本社

〒136-0071 東京都江東区亀戸 1-5-7 日鐵NDタワー

電話： (03) 5627-2059 オートメーショングループ

ファックス： (03) 5627-2055

電子メール： io-system-jp@wago.com

この取扱説明書において使用される会社名、ソフトウェアおよびハードウェアの名称は、一般的に、商標法または特許法により保護されています。

目次

1 重要事項	1
1.1 法的原則	1
1.1.1 著作権	1
1.1.2 使用者の資格基準	1
1.1.3 750 シリーズ使用の準拠規格	1
1.1.4 デバイスの取扱い上の制約	1
1.2 750 シリーズの取扱いに関する規格および規制	2
1.3 注意記号	2
1.4 安全上の注意	3
1.5 使用書体の説明	4
1.6 記数法	4
1.7 適用範囲	5
1.8 略号	5
2 WAGO I/O システム 750	6
2.1 システム概要	6
2.2 技術仕様	7
2.3 製造番号	13
2.4 製品のバージョンアップ	14
2.5 保管、アセンブリ、輸送	14
2.6 機械的セットアップ	15
2.6.1 インストール位置	15
2.6.2 全長	15
2.6.3 キャリアレールへの取り付け	16
2.6.4 スペース	17
2.6.5 モジュールの着脱	18
2.6.6 組立順序	19
2.6.7 内部バスとデータ接点	20
2.6.8 電源接点	21
2.6.9 電線接続	22
2.7 電源	23
2.7.1 電氣的分離	23
2.7.2 システム電源	24
2.7.3 フィールド電源	27
2.7.4 電源に関する補助的な規則	31

2.7.5	電圧供給例	32
2.7.6	電源ユニット	33
2.8	接地	34
2.8.1	DINレールの接地	34
2.8.2	接地機能	35
2.8.3	保護接地	36
2.9	シールドディング（スクリーニング）	37
2.9.1	一般事項	37
2.9.2	通信バスケーブル	37
2.9.3	信号線	37
2.9.4	ワゴシールド（スクリーン）結線システム	38
2.10	アセンブリのガイドラインおよび規格	38
3	フィールドバスコントローラ	39
3.1	フィールドバスコントローラ 750-837、750-838	39
3.1.1	概要	39
3.1.2	適合性	40
3.1.3	ハードウェア	41
3.1.4	オペレーティングシステム	47
3.1.5	プロセスイメージ	49
3.1.6	データ交換	50
3.1.7	WAGO-I/O-PRO CAAによるPFCのプログラミング	63
3.1.8	CANopenフィールドバスノードの立ち上げ	68
3.1.9	LED表示	73
3.1.10	技術データ	79
4	CANopen	81
4.1	概要	81
4.2	ネットワークアーキテクチャ	82
4.2.1	送信媒体	82
4.2.2	ケーブル接続	83
4.2.3	ネットワークトポロジ	86
4.2.4	インタフェースモジュール	87
4.2.5	コンフィグレーションソフトウェア	87
4.3	ネットワーク通信	88
4.3.1	通信オブジェクト	89
4.3.2	CANopenフィールドバスコントローラの通信状態	100
4.3.3	ネットワーク管理オブジェクト	102

4.3.4	オブジェクトディクショナリ	105
4.3.5	PDO送信	143
4.3.6	SYNC監視	148
4.3.7	ノードガーディング	148
4.3.8	ハートビート監視	149
4.3.9	エラーメッセージ（エマージェンシ）	150
5	I/Oモジュール	154
5.1	概要	154
5.1.1	デジタル入力モジュール	154
5.1.2	デジタル出力モジュール	155
5.1.3	アナログ入力モジュール	156
5.1.4	アナログ出力モジュール	157
5.1.5	特殊モジュール	157
5.1.6	システムモジュール	158
5.2	CANopen用プロセスデータ構造	159
5.2.1	デジタル入力モジュール	159
5.2.2	デジタル出力モジュール	161
5.2.3	アナログ入力モジュール	165
5.2.4	アナログ出力モジュール	166
5.2.5	特殊モジュール	167
5.2.6	システムモジュール	180
6	爆発性環境での使用について	181
6.1	はじめに	181
6.2	保護対策	181
6.3	CENELECおよびIECに基づく分類	181
6.3.1	区 分	181
6.3.2	防爆グループ	182
6.3.3	装置カテゴリー	183
6.3.4	温度等級	183
6.3.5	着火保護のタイプ	184
6.4	NEC 500 に基づく分類	185
6.4.1	区 分	185
6.4.2	防爆グループ	185
6.4.3	温度等級	186
6.5	識別（ラベリング）	187
6.5.1	欧 州	187

6.5.2	北 米	188
6.6	設置規制	189

1 重要事項

本章では、個々の節で触れている最も重要な安全上の要求や、注意事項の要点のみを掲げます。健康を守り、装置への損傷を防ぐために、安全ガイドラインをよく読んで遵守することが肝心です。

1.1 法的原則

1.1.1 著作権

本書は図表を含めてすべて著作権で保護されています。本書に明記された著作権条項に抵触する使用は禁じられています。複製、翻訳、電子的手段または複写による保存および修正を行うには、ワゴコンタクトテクニク社（ドイツ）の同意書が必要です。これに違反した場合、当社には損害賠償を請求する権利が生じます。

ワゴコンタクトテクニク社（ドイツ）は、技術の進展に合わせて改変を行う権利を保有します。特許または実用新案による法的保護を受けている場合、ワゴコンタクトテクニク社（ドイツ）はすべての権利を保有します。なお、他社製品については、常にそれらの製品名の特許権について記載しません。ただし、それらの製品に関する特許権等を除外するものではありません。

1.1.2 使用者の資格基準

本書で説明する製品を使用するためには、次の表で示されるように、各作業内容に応じて特別な資格が必要となります。

作業内容	電気担当者	教育受講者*)	専門技術者**) (PLC プログラミングの資格所有)
組み立て	可	可	
コミッショニング	可		可
プログラミング			可
メンテナンス	可	可	
トラブルシューティング	可		
分解	可	可	

*)教育受講者は資格所有者または電気担当者からトレーニングを受けた人です。

**)専門技術者は技術トレーニング、知識、経験を通じて関連した基準を満たし、各作業内容の項目で起こりうる危険を識別できる能力がある人です。

全てのユーザは適用されている規格に熟知していなければなりません。

不適切な作業による損害、または本書の内容を順守しないために発生したワゴ製品および他社製品の損害について、ワゴコンタクトテクニク社（ドイツ）は一切の責任を負いかねますのでご了承ください。

1.1.3 750 シリーズ使用の準拠規格

モジュラ式 I/O システム 750 のバスカプラやコントローラは、I/O モジュールからデジタルおよびアナログ信号を受信し、それをアクチュエータまたは上位の制御システムに送信します。ワゴのコントローラを用いれば、信号を前処理することもできます。

750 シリーズのデバイスは IP20 保護クラスで設計されています。指先が触れないよう、また直径が最大 12.5mm の固形物が入らないように保護されています。他に規定がない限り、デバイスは湿気や埃のある環境で使用してはいけません。

1.1.4 デバイスの取扱い上の制約

製品は個々の用途に応じて、専用のハードウェアおよびソフトウェアを構成した上で工場か

2 重要事項

750 シリーズの取扱いに関する規格および規制

ら出荷されます。ハードウェア、ソフトウェアおよびファームウェアの変更は、説明書に書かれている可能性の範囲内でのみ許されます。ハードウェアまたはソフトウェアの全ての変更、および部品の規格を遵守しない使用に対しては、ワゴコンタクトテクニク社（ドイツ）は一切の責任を負いかねますのでご了承ください。

新規のハードウェアやソフトウェア、またはその変更に関する要求は全て、ワゴコンタクトテクニク社（ドイツ）に直接連絡するようお願いいたします。

1.2 750 シリーズの取扱いに関する規格および規制

設置の際は関連した規格や規制を遵守してください。

- データおよび電源線は、設置の際の障害を避けるため、また人への危険を除くために規格に則って配線や設置をしてください。
- 設置、立ち上げ、メンテナンスおよび修理をするときは、お使いの機械の事故防止規制（例：BGV A3「電気設置および機器」）を守ってください。
- 緊急停止の機能や装置は“無効”に設定してはいけません。関連規格を参照してください（例：DIN EN418）。
- 設置をするときは、電磁障害を除去できるように、EMC ガイドラインに従って行ってください。
- 追加的な手段がない家庭環境で 750 シリーズを動かす場合は、EN61000-6-3 に準拠した放射制限（電磁障害、放射）の条件を満たしている場合のみ使用が許されます。関連情報は 2.2 章「テクニカルデータ」に記述されています。
- DIN EN61340-5-1/-3 に基づいて静電放電に対する安全対策を遵守してください。モジュールを取扱うときは、周囲環境（人、作業場、梱包）の接地が十分に行われていることを確認してください。
- 制御盤の設置に関しては、関連する有効で、かつ適用可能な規格を守ってください。

1.3 注意記号



危険

傷害防止のため、指示内容を必ず順守してください。



警告

装置の損傷防止のため、指示内容を必ず順守してください。



注意

円滑な動作を確保するため、限界条件を必ず守ってください。



静電気（ESD）

静電放電によって損傷する恐れのある部品を示します。部品を扱う際には予防対策を行ってください。



メモ

問題を起こさずに、装置の効率的な動作を保証するために守るべき重要なメモです。



詳細情報

本書以外の文書、マニュアル、データシート、および Web サイトに関する参照情報です。

1.4 安全上の注意

機械の設置時や動作中にデバイスを装置に接続するときは、以下の安全上の注意を守らなければなりません。



危険

ワゴ I/O システム 750 とそのモジュールは外気にさらされています。組立作業は必ずハウジング、キャビネット、または電気作業室にて行ってください。また、組立場所を鍵またはツールで保護し、許可された有資格者以外の入室を禁じます。



危険

設置、修理、またはメンテナンス作業は、必ずデバイスの電源を全て切ってから行ってください。



注意

故障または損傷したデバイスやモジュール（例：接点の変形の場合など）は、フィールドバスにおいてこの局の機能が長期的には保証できないので交換してください。



注意

モジュールは、浸透性および絶縁性をもつ物質に対しての耐性はありません。そのような物質には、エアロゾル、シリコン、トリグリセリド（ハンドクリームなどに使用される）などがあります。

この種の物質をモジュールの周辺から排除できない場合には、次のような対策が必要になります。

- モジュールを適切なハウジングに収容する
- モジュールを扱うときは必ず清浄な工具または材料を使用する



注意

接点が汚損した場合は、油分を含まない圧縮空気を使用するか、またはエチルアルコールと革布で清掃します。

接点用スプレーは使用しないでください。最悪の場合、接点部分の機能が損なわれます。



注意

デバイスを損傷する恐れがありますので、データや電源線の極性を逆にして接続しないようにしてください。



静電気（ESD）

デバイスは、触れたとき、内部の電子部品が静電放電によって破損する場合があります。

1.5 使用書体の説明

パス名とファイル名は、イタリックで表します。

例： *C:\¥programs¥WAGO-IO-CHECK*

メニュー項目は、太字のイタリックで表します。

例： ***Save***

連続したメニュー項目は、メニュー名の間に¥を記します。

例： File¥New

ボタンは、太字のsmallキャピタルで表します。

例： **ENTER**

キー類は太字で表記し、山括弧で囲みます。

例： **<F5>**

プログラムコードは、Courier フォントで表記します。

例： END_VAR

1.6 記数法

記数法	例	備考
10 進	100	通常の表記法
16 進	0x64	C での表記法
2 進	'100' '0110.0100'	「'」で囲む 4 ビットごとにドットで区切ります。

1.7 適用範囲

本マニュアルは、CANopen 用フィールドバスコントローラおよび共通モジュールのワゴ I/O システムについて記載したものです。

型 番	説 明
750-837	CANopen フィールドバスコントローラ、10k ボー～1M ボー デジタル/アナログ信号対応、バス接続=MCS
750-837/020-000	CANopen フィールドバスコントローラ、10k ボー～1M ボー デジタル/アナログ信号対応、バス接続=MCS メモリ容量：256kB プログラム、192kB データ RAM
750-837/021-000	CANopen フィールドバスコントローラ、10k ボー～1M ボー デジタル/アナログ信号対応、バス接続=MCS メモリ容量：640kB プログラム、832kB データ RAM
750-838	CANopen フィールドバスコントローラ、10k ボー～1M ボー デジタル/アナログ信号対応、バス接続=D-Sub
750-838/020-000	CANopen フィールドバスコントローラ、10k ボー～1M ボー デジタル/アナログ信号対応、バス接続=D-Sub メモリ容量：256kB プログラム、192kB データ RAM
750-838/021-000	CANopen フィールドバスコントローラ、10k ボー～1M ボー デジタル/アナログ信号対応、バス接続=D-Sub メモリ容量：640kB プログラム、832kB データ RAM

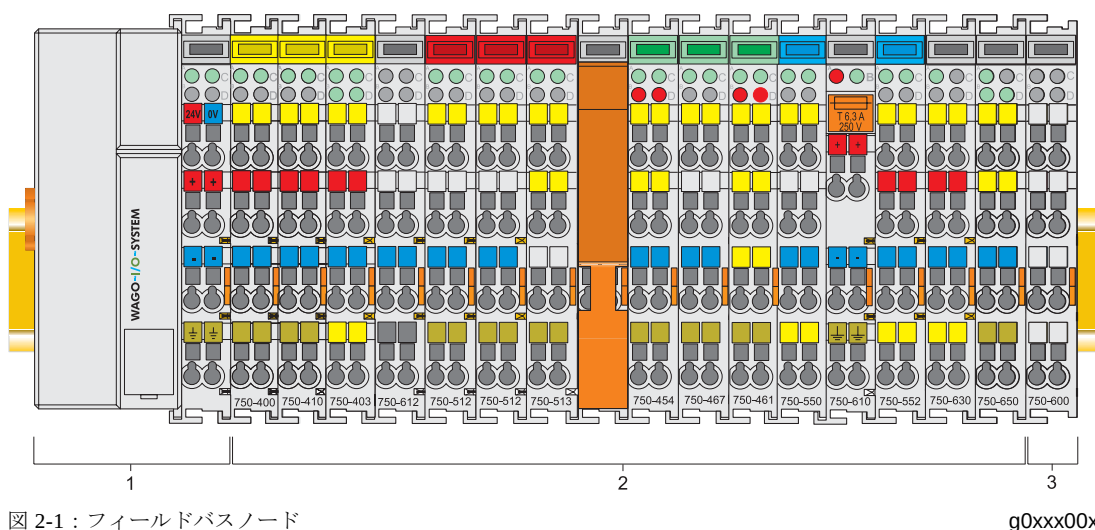
1.8 略号

AI	アナログ入力
AO	アナログ出力
BC	バスコントローラ
CAL	CAN Application Layer (CAN アプリケーションレイヤ)
CAN	Control Area Network
COB ID	通信オブジェクト識別子
DI	デジタル入力
DO	デジタル出力
EMCY	エマージェンシ・オブジェクト
I/O	入力/出力
ID	識別子、識別
Idx	インデックス
M	マスタ
NMT	ネットワークマネジメント
PDO	Process Data Object
RO	Read Only (リードオンリー)
RTR	Remote Transmit Request (リモート送信要求)
RxPDO	受信 PDO
RW	Read/Write (リード/ライト)
SDO	Service Data Object
S-Idx	サブインデックス
TxPDO	送信 PDO

2 ワゴ I/O システム 750

2.1 システム概要

ワゴ I/O システム 750 は、どんなフィールドバスにも使用できるモジュール式 I/O システムです。本製品は、(1) フィールドバスカプラ／コントローラと、(2) あらゆる信号に対応するフィールドバスモジュール（最大 64 枚が接続可能）によって構成されます。この組み合わせによってフィールドバスノードが形成されます。ノードの終端には (3) 終端モジュールを使用します。



フィールドバスカプラ／コントローラとしては PROFIBUS、INTERBUS、Ethernet TCP/IP、CAN (CANopen、DeviceNet)、MODBUS、LON などのフィールドバスシステムに対応するバスカプラ／コントローラが用意されています。

バスカプラ／コントローラには、フィールドバスインタフェース、電子回路、および電源端子が設けられています。フィールドバスインタフェースは、フィールドバスに対する物理的インタフェースを形成します。電子回路はバスモジュールのデータ処理を行い、フィールドバス通信に使用できる形に変換します。24V のシステム電源および 24V のフィールド電源は、設けられた電源端子を通じて供給されます。フィールドバスコントローラは、対応したフィールドバスを介して通信します。PFC（プログラマブルフィールドバスコントローラ）により、PLC に対して追加的な機能を実行できます。プログラミングは、WAGO-I/O-PRO 32 または WAGO-I/O-PRO CAA プログラミングツールを使用し、IEC 61131-3 に基づいて行います。

バスカプラ／コントローラには、デジタルおよびアナログの各種 I/O モジュールおよび特殊機能に対応したバスモジュールを接続することができます。バスカプラ／コントローラとバスモジュール間の通信は、内部バスを通じて行われます。

ワゴ I/O システム 750 には、LED による明瞭なチャンネル表示、挿入式のミニ WSB マーカ、および引出式のグループマーカキャリアがあります。アース線接続用の端子を備えたモジュールは 3 線式のセンサ／アクチュエータに直接配線できます。

2.2 技術仕様

機械的データ	
材 質	ポリカーボネート、ポリアミド 6.6
寸法 W×H×L – バスカプラ/コントローラ – I/O モジュール (シングル) – I/O モジュール (ダブル)	– 51mm×65*mm×100mm – 12mm×64*mm×100mm – 24mm×64*mm×100mm * DIN 35 レールの上端からの測定値
インストール方式	インターロックつき DIN 35 レール
モジュール方式	スライドキーとダブテールの二重型
取付け位置	制限なし
マーキング	247 シリーズおよび 248 シリーズのマーキングラベル マーキングラベル用紙は 8×47mm
接 続	
接続方式	ケージクランプ®接続
電線サイズ	0.08～2.5mm ² 、AWG 28-14
電線むき長さ	750 シリーズモジュール使用：8～9m 753 シリーズモジュール使用：9～10mm
接 点	
電源ジャンパー接点	ブレード接点／ばね接点 セルフクリーニング機構
電源端子経由の最大電流	10 A
I _{max} における電圧降下	モジュール 64 枚につき 1 V 未満
データ接点	スライド接触、硬質金めっき 1.5μ、セルフクリーニング
気候環境条件	
動作温度	0～55℃ -20℃～+60℃：温度拡張モジュール (750-xxx/025-xxx) 使用の場合
保管温度	–20～+85℃
相対湿度	5%～95% (結露がないこと)
有害物質への耐性	IEC 60068-2-42 および IEC 60068-2-43 に準拠
汚染ガス濃度 (相対湿度 75%以下)	SO ₂ ≤25ppm H ₂ S ≤10ppm
特別条件	以下に該当する環境では追加的な対策を実施してモジュールを保護すること – ダスト、腐食性蒸気またはガス – 電離放射

安全な電氣的絶縁				
空間絶縁距離と沿面距離		IEC 60664-1 に準拠		
IEC61131-2 に準拠した汚染度		2		
保護等級				
保護等級		IP 20		
電磁環境適合性（EMC）＊				
EN 61000-6-2（2001）準拠の工業環境電磁妨害イミュニティ				
試験規格	試験値		強度等級	評価基準
EN 61000-4-2 ESD	4kV/8kV（接点/大気）		2/3	B
EN 61000-4-3 電磁場	10V/m 80MHz～1GHz		3	A
EN 61000-4-4 バースト	1kV/2kV（データ/電源）		2/3	B
EN 61000-4-5 サージ	データ	-/-（ライン/ライン）		B
		1kV（ライン/アース）	2	
	DC 電源	0.5kV（ライン/ライン）	1	B
		0.5kV（ライン/アース）	1	
	AC 電源	1kV（ライン/ライン）	2	B
		2kV（ライン/アース）	3	
EN 61000-4-6 RF イミュニティ	10V/m 80% AM（0.15～80MHz）		3	A
EN 61000-6-4（2001）準拠の工業環境電磁妨害エミッション				
試験規格	制限値/QP＊		周波数範囲	測定距離
EN 55011（AC 電源、伝導ノイズ）	79dB（μV）		150kHz～500kHz	
	73dB（μV）		500kHz～30MHz	
EN 55011（放射ノイズ）	40dB（μV/m）		30MHz～230MHz	10m
	47dB（μV/m）		230MHz～1GHz	10m
EN 61000-6-3（2001）準拠の住宅環境電磁妨害エミッション				
試験規格	制限値/QP＊		周波数範囲	測定距離
EN 55022（AC 電源、伝導ノイズ）	66～56dB（μV）		150kHz～500kHz	
	56 dB（μV）		500kHz～5MHz	
	60 dB（μV）		5MHz～30MHz	
EN55022（DC 電源、伝導ノイズ）	40～30 dB（μA）		150kHz～500kHz	
	30 dB（μA）		500kHz～30MHz	
EN55022（放射ノイズ）	30 dB（μV/m）		30MHz～230MHz	10m
	37 dB（μV/m）		230MHz～1GHz	10m

*) Quasi Peak

IEC 61131-2 準拠の機械強度		
試験規格	周波数範囲	制限値
IEC 60068-2-6 耐振動性	$5\text{Hz} \leq f < 9\text{Hz}$	1.75mm 振幅（連続） 3.5mm 振幅（短時間）
	$9\text{Hz} \leq f \leq 150\text{Hz}$	0.5g（連続） 1g（短時間）
	振動試験条件は以下のとおり a) 振動適用手順 毎分 1 オクターブの変化率で掃引 b) 試験方向 互いに直角の 3 軸方向の各軸で掃引	
IEC 60068-2-27 耐衝撃性		15g
	衝撃試験条件は以下のとおり a) パルスの種類・・・正弦半波 b) パルス強度 保持時間 11ms c) 互いに直角の 3 軸方向の各軸で正負両方向に連続 3 回の衝撃を付加	
IEC 60068-2-32 自由落下		≦ 1m（初期包装状態のモジュール）



メモ

モジュールの技術仕様が本章で記述されている値と異なっている場合、個々のモジュールのマニュアルに記載されている技術仕様が有効となります。

船舶規格が承認されたワゴ I/O システム 750 製品に対しては、以下のような補足ガイドラインが適用されます。

電磁環境適合性 (EMC) *			
ドイツ・ロイド船級協会規格 (2003) に基づいた電磁妨害イミュニティ			
試験規格	試験値	強度等級	評価基準
IEC 61000-4-2 ESD	6kV/8kV (接点/大気)	3/3	B
IEC 61000-4-3 電磁場	10V/m 80MHz～2GHz	3	A
IEC 61000-4-4 バースト	1kV/2kV (データ/電源)	2/3	A
IEC 61000-4-5 サージ	AC/DC 電源	0.5kV (ライン/ライン)	1
		1kV (ライン/アース)	2
EN 61000-4-6 RF イミュニティ	10V/m 80% AM (0.15～80MHz)	3	A
タイプテスト、AF イミュニティ (調波)	3V、2W	-	A
タイプテスト、高電圧	DC 755V AC 1500V	-	-
ドイツ・ロイド船級協会規格 (2003) に基づいた電磁妨害エミッション			
試験規格	制限値	周波数範囲	測定距離
タイプテスト (EMC1、伝導) ブリッジ制御に適用	96～50dB (μ V)	10kHz～150kHz	
	60～50dB (μ V)	150kHz～350kHz	
	50dB (μ V)	350kHz～30MHz	
タイプテスト (EMC1、放射) ブリッジ制御に適用	80～52dB (μ V/m)	150kHz～300kHz	3m
	52～34dB (μ V/m)	300kHz～30MHz	3m
	54dB (μ V/m)	30MHz～2GHz	3m
	24dB (μ V/m)	156MHz～165MHz	3m
ドイツ・ロイド船級協会規格 (2003) に基づいた機械強度			
試験規格	周波数範囲	制限値	
IEC 60068-2-6 耐振動性 (カテゴリ A～D)	2Hz≤f<25Hz	±1.6mm 振幅 (連続)	
	25Hz≤f<100Hz	4g (連続)	
	振動試験条件 a) 周波数変化: 最大 1 オクターブ/分 b) 振動方向: 3 方向		

適用範囲	電磁妨害エミッション 要求規格	電磁妨害イミュニティ 要求規格
工業地域	EN 61000-6-4 (2001)	EN 61000-6-2 (2001)
住宅地域	EN 61000-6-3 (2001) *	EN 61000-6-1 (2001)

*) 以下のフィールドバスカプラ／コントローラをインストールしたシステムは、住宅地での妨害電波の放出に対する要求事項を満たします。

ETHERNET 750-342/-841/-842
LonWorks 750-319/-819
CANopen 750-337/-837
DeviceNet 750-306/-806
MODBUS 750-312/-314/ -315/ -316
750-812/-814/ -815/ -816

特別な許可を受けると、このシステムは、他のフィールドバスコントローラ／コントローラと共に住居地域（住宅地、商業地、中小企業）で使用できます。特別な許可は、所轄機関または検査機関から得ることができます。

各モジュールの最大電力消費値は次のとおりです。

モジュールの最大電力消費値	
バスモジュール	0.8W／バスターミナル (全電力消費、システム／フィールド)
フィールドバスカプラ／ コントローラ	2.0W／バスカプラ／コントローラ



警告

インストールした全モジュールに対する電力消費は、ハウジング（キャビネット）が通電できる最大電力を超えないものとします。

ハウジングの寸法を決める際には、外部温度が高くてもハウジング内の温度が許容周囲温度の 55℃を超えることがないように考慮してください。

寸 法

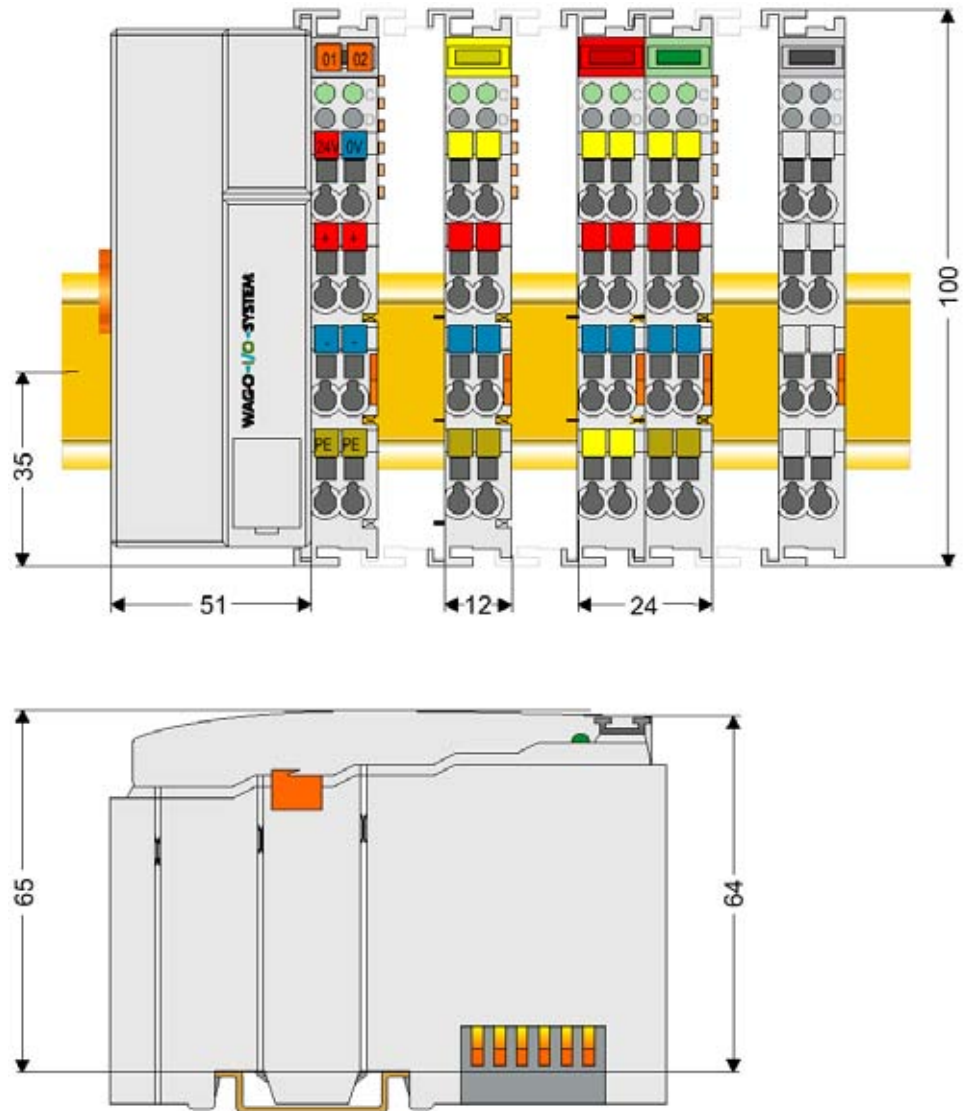


図 2-2 : 寸法

g1xxx05e

2.3 製造番号

製造番号は、生産直後の出荷時期を示しています。この番号はモジュールの側面に印刷されています。

製造番号はまた、フィールドバスコントローラまたはコントローラ上のコンフィグレーションおよびプログラミング用インタフェースのカバーにも印刷されています。

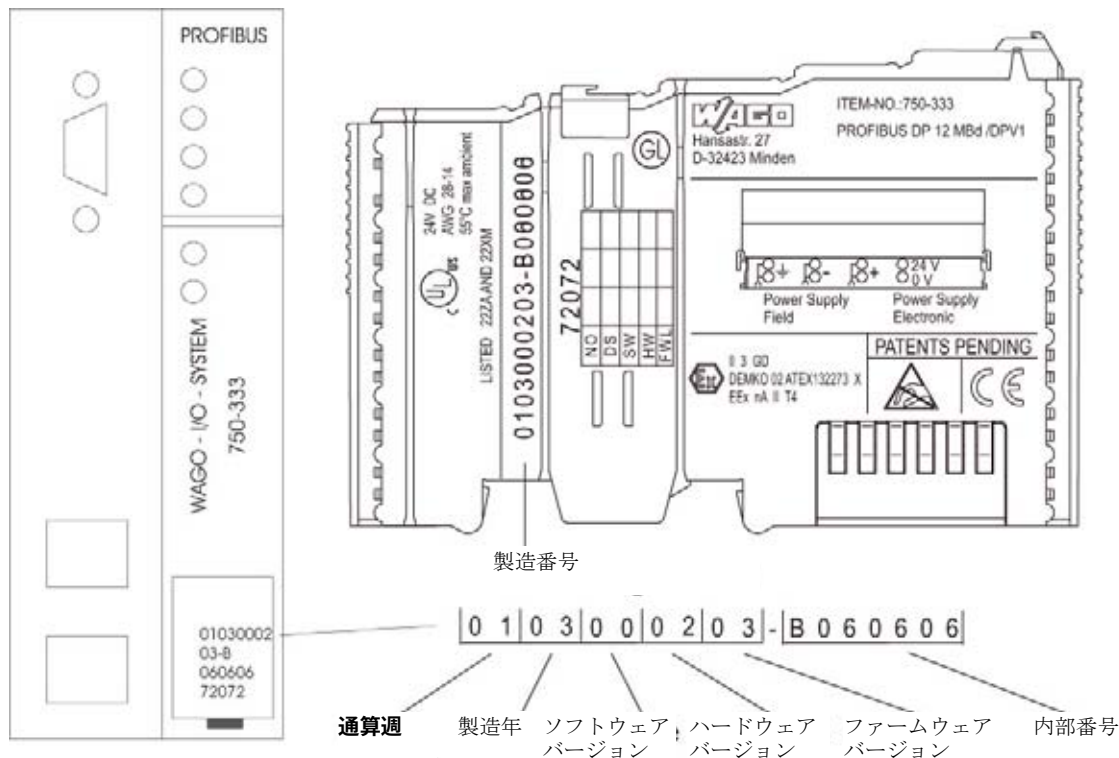


図 2-3 : 製造番号

g1xxx09e

製造番号は、通算製造週、製造年、ソフトウェアバージョン（バージョン番号がある場合）、コントローラのハードウェアバージョン、ファームウェアローダのバージョン（バージョン番号がある場合）、ならびにワゴコンタクトテクニック社（ドイツ）用内部情報で構成されます。

2.4 製品のバージョンアップ

製品にバージョンアップがあった場合の履歴を記すために、各モジュールの側面にはバージョンアップ表が予め印刷されています。

この表には過去 3 回までのバージョンアップが登録でき、以下の項目があります。バージョンアップがあったとき該当欄に番号などが印刷されます。

	1 回目	2 回目	3 回目	
生産番号	NO			
更新日	DS			
ソフトウェアバージョン	SW			
ハードウェアバージョン	HW			
ファームウェアローダバージョン	FWL			カブラ/コントローラのみ

バスカブラやコントローラの場合は、コンフィグレーション・プログラミングインタフェースのカバーにも更新された製造番号が印刷されます。

元の製造番号は、モジュールの側面に残されています。

2.5 保管、アセンブリ、輸送

製品は、可能な限り初期パッケージに入れて保管します。初期パッケージは輸送時にも製品を最適な状態で保護します。

製品をアセンブリまたは再包装する際は、接点を汚損または損傷しないように注意してください。製品は適切な容器に格納または包装して保管および輸送します。その際、静電気対策を考慮してください。

アミン、アミド、およびシリコンの汚損防止用として、裸の製品の輸送には金属コーティングを施した静電遮蔽輸送袋（例：3M 1900E）を使用します。

2.6 機械的セットアップ

2.6.1 インストール位置

水平方向や垂直方向をはじめ、どのような方向にもインストール可能です。



注 意

垂直アセンブリの場合、安全対策としてスリップ防止用のエンドストップを取り付けることが必要です。

WAGO 型番 249-116 DIN 35 レール用 6mm 幅エンドストップ

WAGO 型番 249-117 DIN 35 レール用 10mm 幅エンドストップ

2.6.2 全長

ノードの最大全長は次のように計算します。

数量	幅	モジュール
1	51mm	バスカブラ/コントローラ
64	12mm	バスモジュール - 入出力 - 電源入力モジュール - その他
1	12mm	終端モジュール
合計		831mm



警告

ノードの最大全長が 831mm を超えないようにしてください。

2.6.3 キャリアレールへの取り付け

2.6.3.1 キャリアレールの特性

すべてのシステムモジュールは、欧州規格 EN 50022 (DIN 35) に準拠したキャリアレールに直接スナップ装着できます。



警告

ワゴは I/O システムにとって最適な標準キャリアレールを提供します。それ以外のキャリアレールを使用するときは、キャリアレールの仕様検査と承認をワゴコンタクトテクニク社 (ドイツ) から受けてください。

キャリアレールの機械的・電氣的属性は種類によって異なります。キャリアレールに対して最適なシステムを設置するには、最低限以下の条件に従うことが必要です。

- 非腐食性の材質であること。
- 大半のモジュールにはキャリアレール用の接点があり、それによって電磁雑音を地面に逃しています。腐食を防止するには、スズめっきのキャリアレール接点がキャリアレール材質との間でガルバニ電池を形成しないことが必要です。そのときに生成される電位差は 0.5V を超えます (20°C、0.3%の食塩水)。
- キャリアレールは、システムの EMC 対策およびバスモジュール結線のシールドを最適な形でサポートする必要があります。
- 十分に安定したキャリアレールを選択し、必要であれば複数のアセンブリ留箇所 (20cm ごと) を用いて湾曲やねじれを防止することが必要です。
- モジュールを安全に保持するため、キャリアレールの外形を変更しないでください。特にキャリアレールを短くして取り付ける場合は、破碎したり曲げたりしないでください。
- モジュールの底部はキャリアレールの形にはまります。高さ 7.5mm のキャリアレールについては、アセンブリ留箇所 (ネジ) をノードの下でリベット止めします (頭に溝が入った非脱落型ネジまたはブラインドリベット)。

2.6.3.2 ワゴ製 DIN レール

ワゴのキャリアレールは、電氣的／機械的要求事項を満たしています。

型番	説明
210-113 /-112	35×7.5 ; 1mm ; 鋼、黄色、クロメート処理済、溝あり／なし
210-114 /-197	35×15 ; 1.5mm ; 鋼、黄色、クロメート処理済、溝あり／なし
210-118	35×15 ; 2.3mm ; 鋼、黄色、クロメート処理済、溝なし
210-198	35×15 ; 2.3mm ; 銅、溝なし
210-196	35×7.5 ; 1mm; アルミ、溝なし

2.6.4 スペース

隣接するモジュール、ケーブルコンジット、ケーシングとフレームの間には、フィールドバスノード全体に対して必要なスペースを確保します。

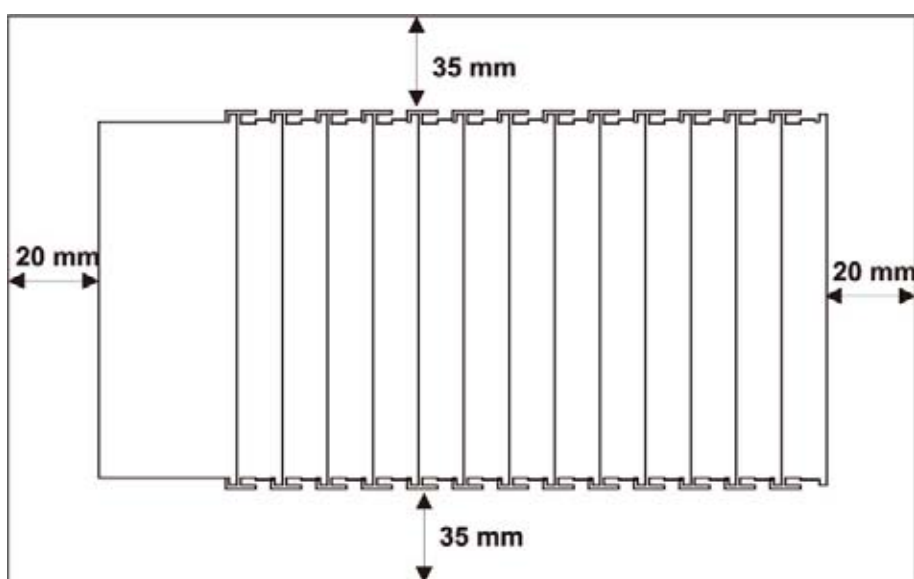


図 2-4 : スペース

g01xx13x

スペースは、熱伝達、絶縁、配線のための空間です。また、ケーブルコンジットとの間のスペースは、電磁干渉による動作妨害の防止にもつながります。

2.6.5 モジュールの着脱



警告

モジュールの着脱作業を開始する前に必ず電源を切ってください。

バスカプラ/コントローラが動いたりすることのないように、ロックディスクを使ってキャリアレールに固定します。ロックディスクの上の溝をドライバで押し込みます。

バスカプラ/コントローラを引き出すには、ロックディスクの下の溝をドライバで押してロックを解除し、オレンジ色のロック解除つまみを引っ張ります。

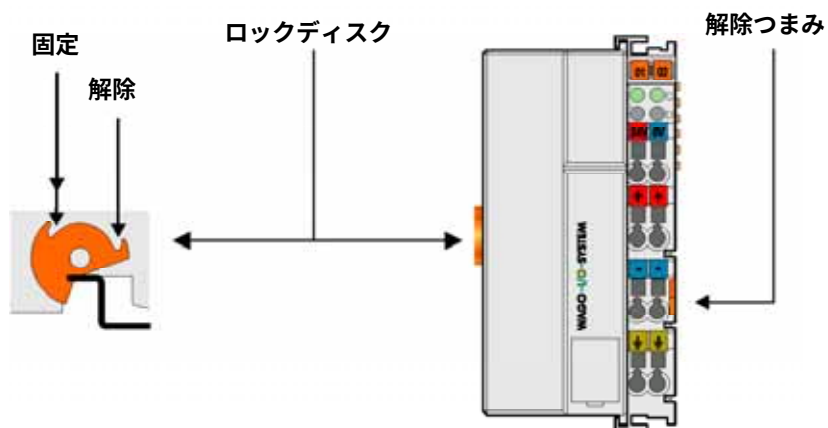


図 2-5 : バスカプラ/コントローラとロック解除つまみ

g01xx12e

個々の I/O モジュールをユニットから引き出すときにも、ロック解除つまみを引っ張ります。

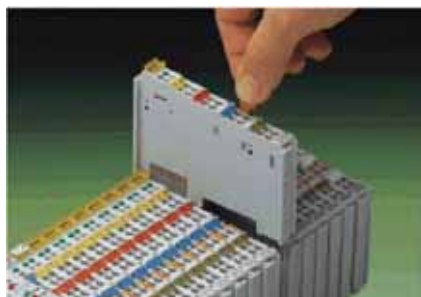


図 2-6 : バスモジュールの取り出し

p0xxx01x



危険

PE を切り離しても人や装置に危険が及ばないことを確認してください。

接地線の環状結線については 2.8.3 節をお読みください。

2.6.6 組立順序

すべてのバスモジュールは、欧州規格 EN 50022 (DIN 35) に準拠したキャリアレールに直接スナップ装着できます。

各モジュールが凹凸形状をしていることにより、信頼度の高い位置決めおよび接続が実現します。自動ロック機能により、個々のモジュールはレールに確実に取付けられます。

バスモジュールは、設計図に基づいて、バスカプラ/コントローラから順に隣接させて接続します。電源接点（オス接点）を備えたバスモジュールの中には電源接点の個数が足りないバスモジュールとは接続できないものがあるので、同電位グループ（電源接点を介した接続）であるかどうかは確認できます。



注意

バスモジュールをバスカプラ/コントローラと接続するときは、必ず上から差し込みます。



警告

バスモジュールは絶対に終端端子側からインストールしないでください。アース接点なしのモジュール（4 チャンネル式デジタル入力モジュールなど）が挿入された場合は、たとえば DI4 において隣の接点との空間絶縁距離および沿面距離が小さくなっています。

フィールドバスノードは必ず終端モジュール（750-600）を使って終端してください。

2.6.7 内部バスとデータ接点

バスカプラ/コントローラとバスモジュール間の通信、およびバスモジュールのシステム電源との通信には、内部バスが使用されます。内部バスには 6 個のデータ接点が装備されています。これらは金のばね接点で、セルフクリーニング方式を採用しています。



図 2-7 : データ接点

p0xxx07x



警告

汚損や傷を防ぐため、I/O モジュールの側面にある金のばね接点に手を触れないでください。



静電気 (ESD)

モジュール内の電子部品は、静電放電によって破損する場合があります。モジュールを扱う際には、作業員、作業場、包装などに対して十分な接地を行ってください。また導電性の部分（金接点など）には手を触れないように注意してください。

2.6.8 電源接点

セルフクリーニング方式の電源接点はバスモジュールの側面にあり、フィールド側の供給電圧を送るのに用いられます。電源接点は接触が保護されたばね接点で、バスカプラ/コントローラおよびバスモジュールの右側にあります。モジュールの左側には、これらに対応するオス形の接点があります。



危険

電源接点は端部が鋭くなっています。モジュールの取り扱いには十分注意してください。



注意

バスモジュールには、電源ジャンパ接点がまったくない、またはわずかな数しか装備されていないものがあります。一部のモジュールでは、オス側の接点を受け入れる溝が上面になく、モジュールを隣接して接続できない場合があります。

電源ジャンパ接点

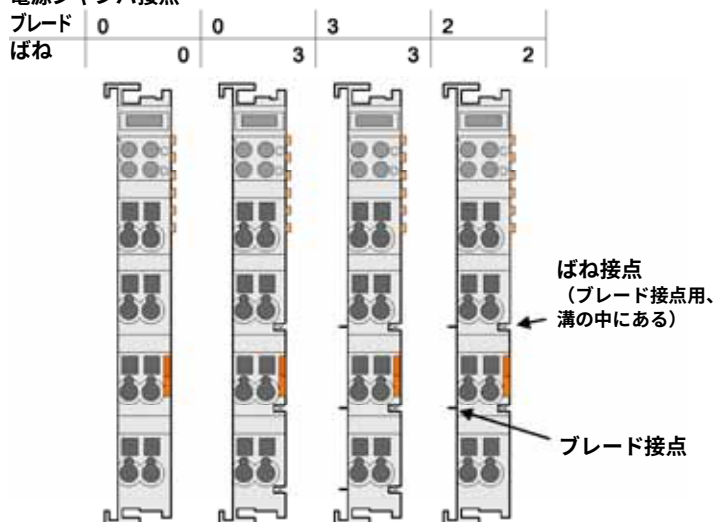


図 2-8 : 電源接点の配置例

g0xxx05e

2.6.9 電線接続

すべてのバスモジュールにはケージランプ®結線金具（スプリング）が装備されています。

ワゴケージランプ®は、単線、撚り線および極細撚り線に適しています。各ランプ箇所は1本の電線を結線できます。

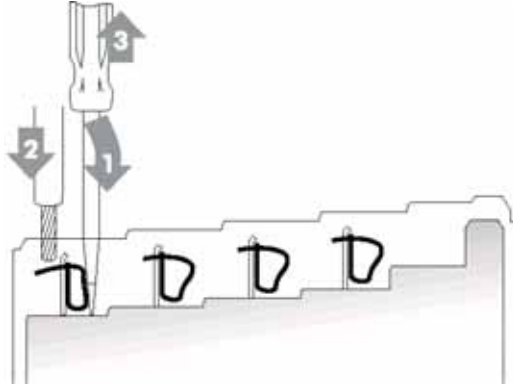


図 2-9 : ケージランプ®による結線

g0xxx08x

ケージランプ®の上の開口部に工具を差し込み、ケージランプ®を開きます。次に開口部分に電線を挿入します。工具を抜くと電線は安全な形で把持されます。

1つのケージランプには1本の電線しか結線できません。1つのケージランプに複数本の電線をつなぐ必要があるときは、ワゴの中継端子を使用して外部配線を行います。

注意

2本の電線を結線する必要がある場合は、フェルールを使用してください。

フェルール：

長さ	8～9mm
最大公称断面積	各 0.5mm ² 、2本合わせて 1mm ²
ワゴ製品	216-103
	または同等の特性をもつ製品

2.7 電源

2.7.1 電氣的分離

フィールドバスノードには電氣的に分離された電圧が 3 種類存在します。

- フィールドバスインタフェースの動作電圧
- バスカブラ/コントローラとバスモジュールの電子回路（内部バス）用電圧
- 内部電子回路（内部バス、ロジック）とフィールド電子回路の間は、すべてのバスモジュールにおいて電氣的に分離されています。一部のアナログ入力モジュールでは、各チャンネルが電氣的に分離されています。詳しくはカタログを参照してください。

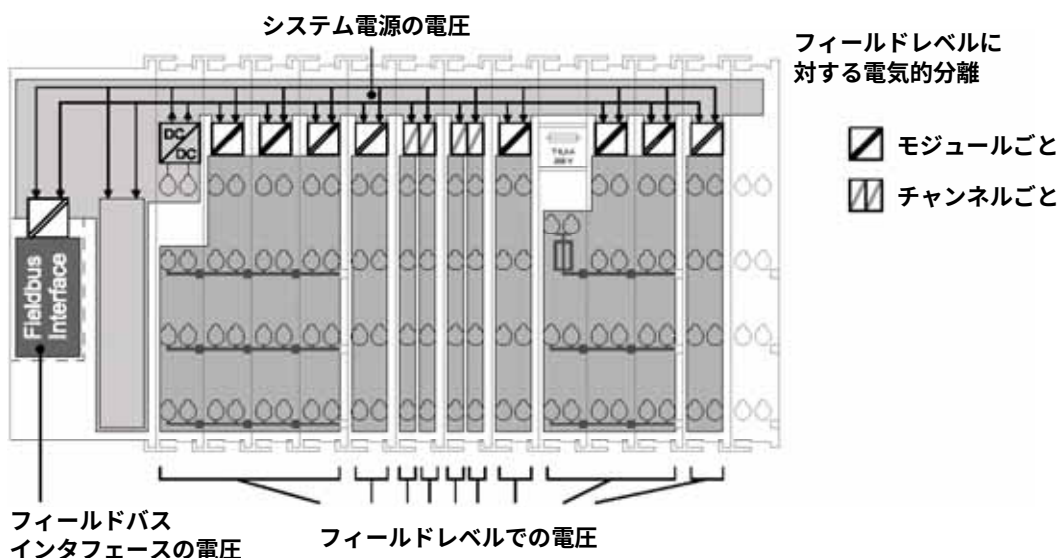


図 2-10：電氣的分離

g0xxx01e



注意

各電圧グループに対して接地線の結線が必要です。保護的導通機能があらゆる状況下で維持されるようにするため、接続は各電圧グループの最初と最後に行ってください（環状結線については 2.8.3 節を参照してください）。それによって、修理点検時にモジュールをノードから取り外した場合でも、実装されたすべてのフィールドデバイスに対して保護的導通接続が保証されます。

24V システム電源と 24V フィールド電源に共通電源装置を使用する場合、その電圧グループに対しては内部バスとフィールドレベルの間の電氣的分離は考慮されません。

2.7.2 システム電源

2.7.2.1 接続

WAGO-I/O-SYSTEM 750 には 24V の直流電源（ -15% または $+20\%$ ）が必要です。電源はバスカプラ/コントローラを通じて供給され、必要であれば内部システム電源入力モジュール（750-613）が補助的に使用されます。電圧供給部には逆電圧保護機能が付いています。

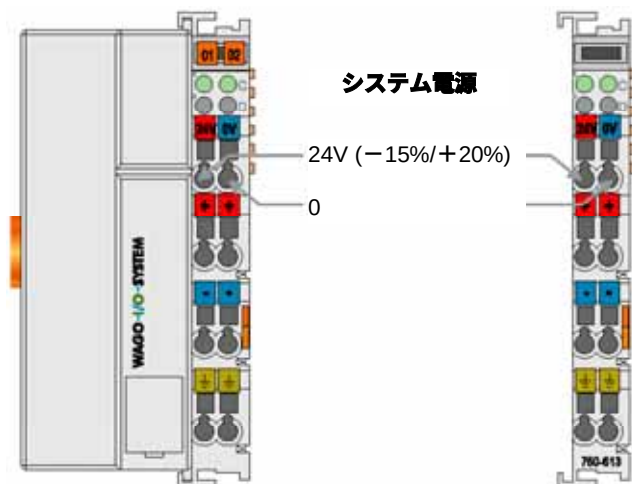


図 2-11 : システム電源

g0xxx02e

直流電流は内部バス（5V システム電圧）を通り、バスカプラ/コントローラの電子回路、フィールドバスインタフェース、およびバスモジュールなど、すべての内部システムモジュールに供給されます。5V のシステム電圧は 24V のシステム電源と電氣的に接続されています。

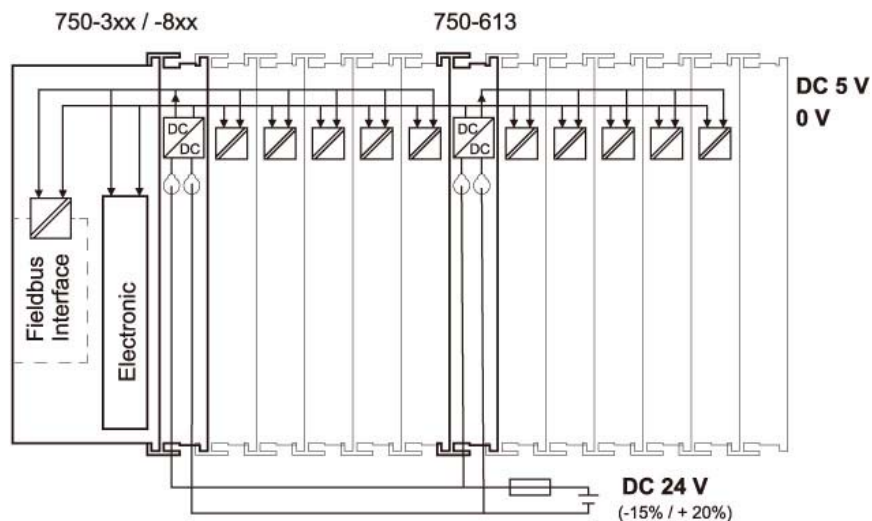


図 2-12 : システム電圧

g0xxx06e



注意

システム電源のオン/オフによるシステムリセットは、すべての電源入力モジュール（バスカプラ/コントローラと 750-613）に対して同時に行う必要があります。

2.7.2.2 モジュール配備

推奨

安定したネットワーク給電がいつでも、どこでも得られるとは限りません。供給電圧の品質を保証するには、安定化電源を使用してください。

バスカプラ/コントローラまたは内部システム電源入力モジュール(750-613)の給電能力は、各モジュールの技術仕様（テクニカルデータ）に記載されています。

内部消費電流*)	バスモジュールおよびバスカプラ/コントローラの電子回路に供給される 5V システム電圧による内部消費電流
バスモジュール用許容残存電流*)	バスモジュールが利用できる電流。バス電源ユニットから供給される。バスカプラ/コントローラおよび内部システム電源入力モジュール（750-613）を参照

*) カタログ W4-Vol.3、取り扱い説明書または WAGO ホームページを参照

例

コントローラ（750-837）：
内部消費電流： 350mA（5V）
許容残存電流
バスモジュール： 1650mA（5V）
合計電流（5V）： 2000mA（5V）

内部消費電流は、各バスモジュールの技術仕様（テクニカルデータ）に記載されています。全体の必要量を計算するには、ノードにインストールされる全バスモジュールの電流値を合計します。



注意

内部消費電流の合計値がバスモジュールへの許容残存電流より大きい場合は、合計消費電流が許容値を超えるモジュール位置の前に内部システム電源入力モジュール（750-613）をインストールする必要があります。

例

CANopen コントローラ（750-837）を使用し、リレーモジュール（750-517）20 枚とデジタル入力モジュール（750-405）20 枚をインストールしたノードの場合：

内部消費電流：
20 × 90mA = 1800mA
10 × 2mA = 20mA
合計 1820mA

コントローラがバスモジュールに対して給電できる量は 1650mA です。したがって、ノードの中央などに内部システム電源入力モジュール（750-613）をインストールする必要があります。

24V システム電源の最大入力電流は 500mA です。正確な消費電流 (I(24V)) は以下の式で計算できます。

コントローラ

$I(5V)_{total} =$ インストールされたバスモジュールの全消費電流 + コントローラの内部消費電流

750-613

$I(5V)_{total}$ = インストールされたバスモジュールの全消費電流

入力電流 $I(24V) = \frac{5V}{24V} \times I(5V)_{total} / \eta$
 $\eta = 0.87$ (公称負荷時の電源効率)



メモ

24V のシステム電源の給電点における消費電流が 500mA を超える場合、その原因としてはノード内のモジュール配備が不適切であるか、モジュールの欠陥が考えられます。

試験時には、すべての出力、特にリレーモジュール使用の際はその出力がアクティブである必要があります。

2.7.3 フィールド電源

2.7.3.1 結線

1～4 線接続方式により、センサおよびアクチュエータがバスモジュールの対応チャンネルに直接結線できます。センサおよびアクチュエータへの給電はバスモジュールが行います。一部のバスモジュールでは、入出力ドライバにフィールド側の供給電圧が必要です。

フィールド側の電源（DC24V）はバスカプラ/コントローラによって供給されます。他の電圧（AC230V など）が必要なときには電源入力モジュールを使用します。また、電源入力モジュールを使用すると各種電圧の供給ができます。結線は 1 つの電源接点について一対で行われます。

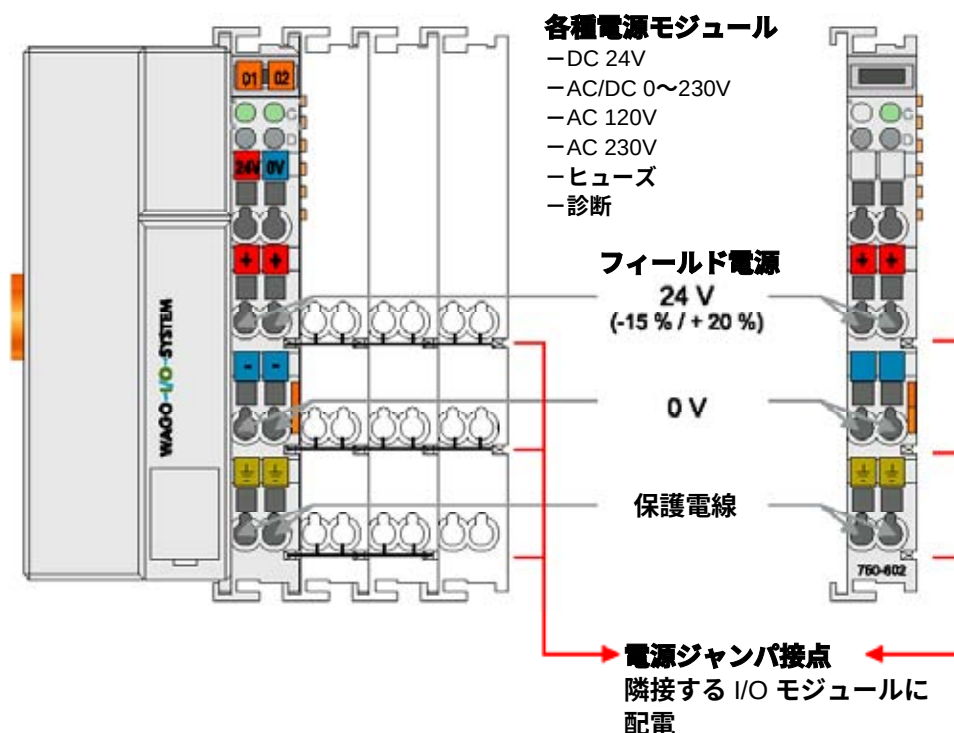


図 2-13 : フィールド給電（センサ/アクチュエータ）

g0xxx03e

フィールド側への供給電圧は、バスモジュールを組み立てたときに電源ジャンパ接点を通して自動的に供給されます。

電源接点の電流負荷が連続して 10A を超えないようにしてください。2 つの接続端子間の電流負荷容量は、接続電線の負荷容量と同じです。

電源入力モジュールを追加すると、電源接点を介したフィールド給電がそこで中断します。そこから新たな給電が行われます。電圧変更の場合も同様です。



注意

バスモジュールには、電源接点がまったくまたはほとんどないものがあります（I/O 機能に依存します）。その場合、対応する給電が中断されます。後続のバスモジュールにおいてフィールド給電が必要な場合は、電源入力モジュールをインストールする必要があります。バスモジュールのデータシートを確認してください。

ノードにおいて複数の電圧を使用する（例：DC24V から AC230V に変更）ときは、スペーサモジュールの使用をお勧めします。電圧を視覚的に分離することで、配線や保守作業時に作業者の注意を促します。配線誤りなどの防止に役立ちます。

2.7.3.2 ヒューズ

適切な電源入力モジュールを選べばフィールド電源に対してヒューズを設けることが各種のフィールド電圧について可能です。

750-601	24V DC	電源／ヒューズ
750-609	230V AC	電源／ヒューズ
750-615	120V AC	電源／ヒューズ
750-610	24V DC	電源／ヒューズ／診断
750-611	230V AC	電源／ヒューズ／診断

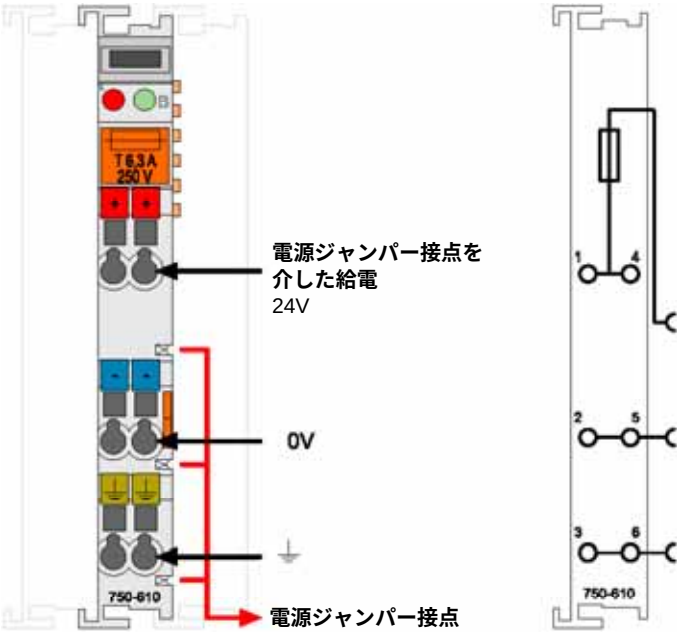


図 2-14：ヒューズキャリアを備えた電源入力モジュール（750-610 の場合）

g0xxx09x



警告

ヒューズキャリアを備えた電源入力モジュールの場合、最大電力損が 1.6W のヒューズ (IEC 127) しか使用できません。

UL 認可システムでは、UL 認可ヒューズ以外は使用しないでください。

ヒューズの挿入や交換、または後続バスモジュールのスイッチオフを行うには、ヒューズホルダを引き出します。これを行うには、たとえばドライバなどを使ってスリット (両側にあります) に引っ掛け、ホルダを引き出します。



図 2-15 : ヒューズキャリアを取り出す

p0xxx05x

横のカバーを引き上げるとヒューズキャリアが開きます。



図 2-16 : ヒューズキャリアを開く

p0xxx03x



図 2-17 : ヒューズを交換する

p0xxx04x

ヒューズを交換した後、ヒューズキャリアを元の位置に戻します。

ヒューズは外部に設置することもできます。WAGOの 281 シリーズと 282 シリーズのヒューズモジュールは、この目的に適しています。



図 2-18 : 自動車用ヒューズに対応したヒューズモジュール (282 シリーズ)

pf66800x



図 2-19 : 回転式ヒューズキャリアを備えたヒューズモジュール (281 シリーズ)

pe61100x



図 2-20 : ヒューズモジュール (282 シリーズ)

pf12400x

2.7.4 電源に関する補助的な規則

WAGO-I/O-SYSTEM 750 は、造船や沿岸または海岸での作業（作業プラットフォーム、荷積み設備など）にも使用できます。このことは、ドイツ船級協会やロイド船級協会などの有力な認定機関の規格への準拠によって証明されています。

規格に沿ったシステム運転を行うには、24V 電源用のフィルタモジュールが必要です。

型番	名称	説明
750-626	電源フィルタ	システム電源およびフィールド電源（24V、0V）用のフィルタモジュール。フィールドバスコントローラ／コントローラおよびバス電源入力モジュール（750-613）向け。
750-624	電源フィルタ	24V フィールド電源（750-602、750-601、750-610）用のフィルタモジュール。

そのため、下に示す給電概念図に従う必要があります。

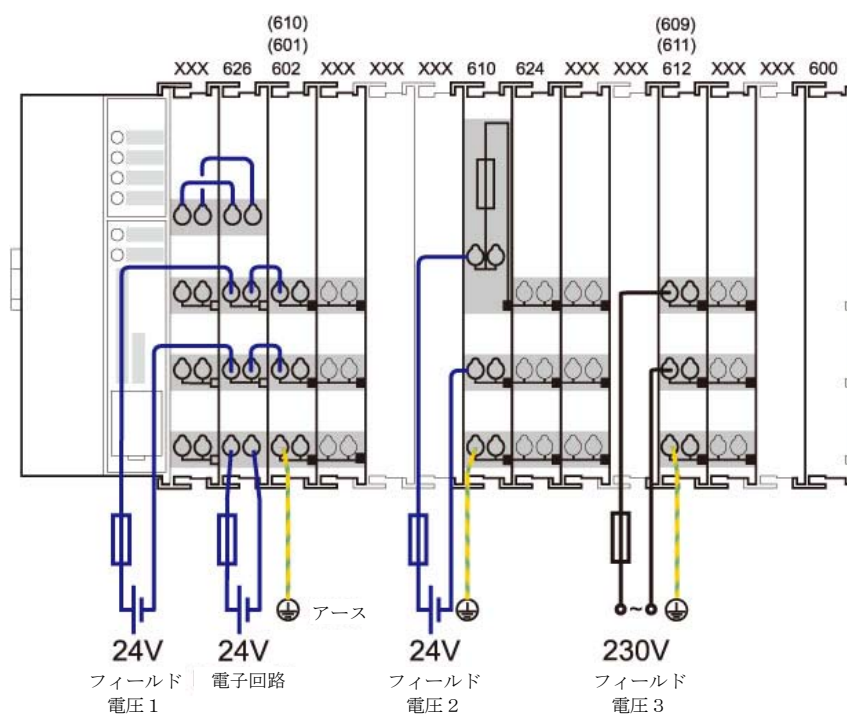


図 2-21：給電概念図

g01xx11e



メモ

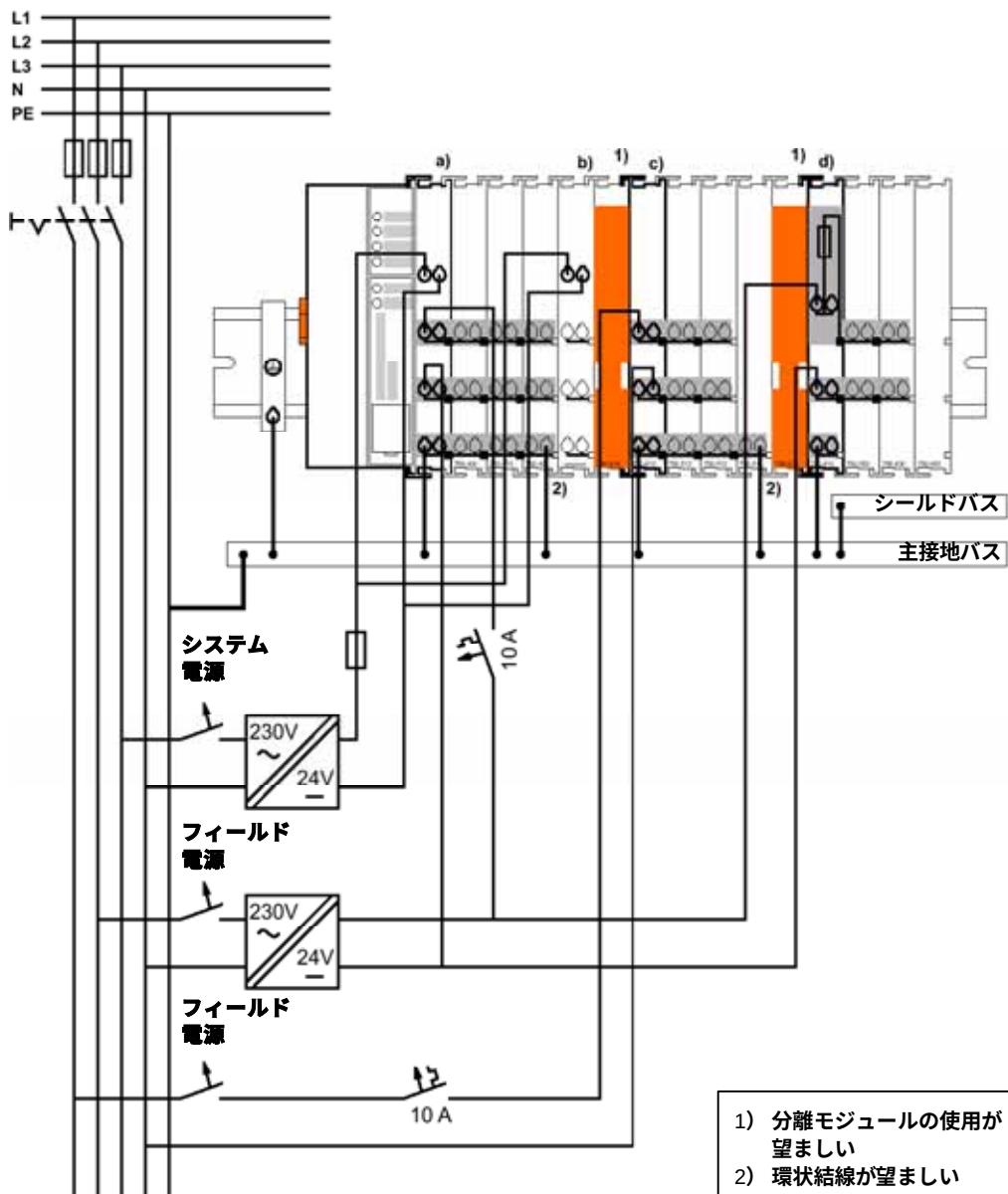
下側の電源接点に保護接地が必要な場合、またはヒューズ保護が必要な場合、追加的な電圧電源ターミナル（750-601/602/610）は必ずフィルタモジュール（750-626）より後で使用する必要があります。

2.7.5 電圧供給例



注意

システム電源とフィールド電源は、アクチュエータ側で短絡が発生してもバス動作に影響が出ないように分離してください。



- 1) 分離モジュールの使用が望ましい
- 2) 環状結線が望ましい
- a) 外部電源モジュールによるカブラ／コントローラの給電
- b) 内部システム電源モジュール
- c) 電源モジュール：パッシブ
- d) 電源モジュール：ヒューズキャリアと診断付き

図 2-22 : 電圧供給例

g0xxx04e

2.7.6 電源ユニット

WAGO-I/O-SYSTEM 750 には 24V の直流システム電源（最大偏差は－15%または＋20%）が必要です。

推奨

安定したネットワーク給電がいつでも、どこでも得られるとは限りません。供給電圧の品質を保証するには、安定化電源を使用してください。

電圧低下が短時間に起きる恐れがある場合はバッファ素子（1A の電流負荷につき 200 μ F）を用意してください。I/O システムのバッファ可能時間は約 1ms です。

フィールド電源に対する電気条件は、給電点ごとに計算します。その際には、フィールド装置とバスモジュールにおける負荷をすべて考慮してください。一部のバスモジュールでは、入出力にフィールド電源の電圧を必要とするため、フィールド電源は後続のバスモジュールにも影響します。



注意

システム電源とフィールド電源は、アクチュエータ側で短絡が発生してもバス動作に影響が出ないように電源回路を分離してください。

ワゴ製品番号	説明
787-612	スイッチング電源、DC24V、2.5A 広い入力電圧範囲 AC90～264V 短絡保護機能付き
787-622	スイッチング電源、DC24V、5A 広い入力電圧範囲 AC90～264V 短絡保護機能付き
787-632	スイッチング電源、DC24V、10A 広い入力電圧範囲 AC85～264V 短絡保護機能付き
288-809 288-810 288-812 288-813	汎用マウントキャリア付きレールマウントモジュール電源 AC 115 V / DC 24 V; 0,5 A AC 230 V / DC 24 V; 0,5 A AC 230 V / DC 24 V; 2 A AC 115 V / DC 24 V; 2 A

2.8 接地

2.8.1 DIN レールの接地

2.8.1.1 フレームアセンブリ

取り付けフレームを組立てるとき、キャリアレールは導電性のキャビネットやハウジングのフレームにネジ止めします。フレームまたはハウジングには接地が必要です。電氣的な接続はネジを通じて行われます。これによってキャリアレールは接地されます。



注意

接地が十分に機能するように、キャリアレールとフレームまたはハウジングとの間には確実な電氣的接続を行ってください。

2.8.1.2 絶縁アセンブリ

構造上、キャビネットのフレームまたは機械部品とキャリアレールとの間に直接の電氣的接続が存在しない場合、アセンブリは絶縁状態になります。この場合、電線によって接地を行ってください。

接地線は、少なくとも 4mm^2 の断面積が必要です。

推奨

金属製の組立プレートとキャリアレールの間で導電接続を行い接地する方法が最も推奨されます。

ワゴのアース端子を使用すると、キャリアレールの個別接地が簡単に行えます。

型番	説明
283-609	単線アース端子台は、キャリアレールに対して自動的に接続を行います。接地線の断面積： $0.2\sim 16\text{mm}^2$ 注：終端・中間プレートもご注文ください（283-320）

2.8.2 接地機能

ワゴ I/O モジュールには接地（アース）機能があり、電磁干渉による外乱を緩和します。I/O システムの一部のモジュールには、電磁気的な外乱をキャリアレールに逃すキャリアレールコンタクトが装備されています。

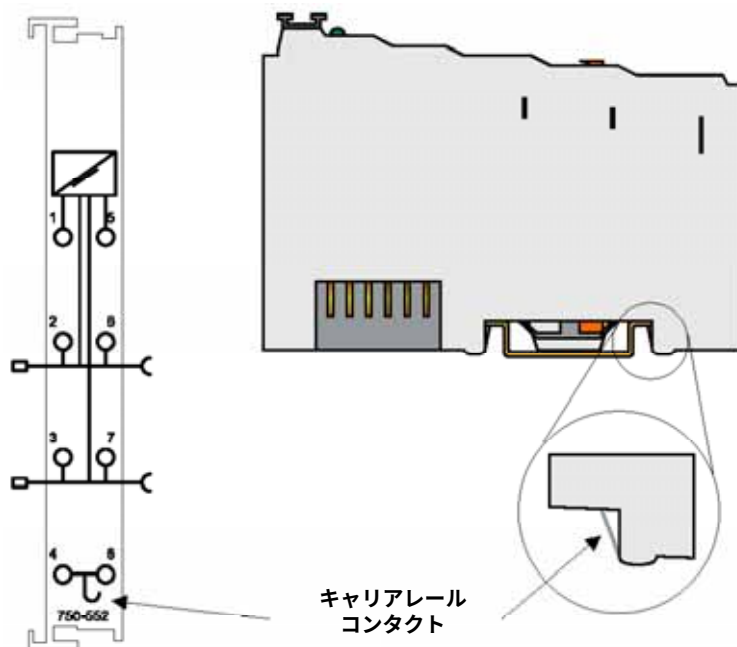


図 2-23 : キャリアレールコンタクト

g0xxx10e



注意

キャリアレールコンタクトとキャリアレールの間には確実な電氣的接続を行ってください。

キャリアレールは接地してください。

キャリアレールの特性については 2.6.3.2 節を参照してください。

2.8.3 保護接地

フィールドレベルでは、接地（アース）線は電源端子の最下部の接続端子に結線され、真横の電源接点を通じて隣接するバスモジュールにつながります。そのバスモジュールにも対応した電源接点があれば、フィールド機器の接地線はそのモジュールの最下部接続端子に直接結線できます。



注意

電源接点による接地線接続がノード内で中断した場合（たとえば 4 チャンネルのバスモジュール）は、再度設置線を配線する必要があります。

接地の環状結線を行うとシステムの信頼性が高まります。バスモジュールが電圧グループから外されたときもアース電位が維持されます。

接地の環状結線を行うときは、接地線を電圧グループの最初と最後に結線します。

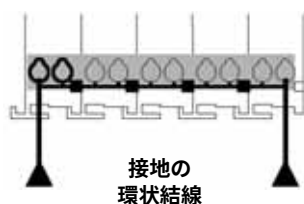


図 2-24 : 環状結線

g0xx07e

2.9 シールディング（スクリーニング）

2.9.1 一般事項

データ線および信号線をシールドすると電磁干渉が減少し、信号品質が高まります。それによって、測定誤差やデータ送受信エラー、場合によっては過電圧による障害まで防止できます。



注意

測定精度に関する仕様を保証するため、シールドは常時実施してください。

データ線および信号線は、すべての高圧ケーブルから離して配線してください。

表面積の大きな部分にはケーブルシールドを施し、アース電位に落とします。
これにより、入力障害を容易に回避できます。

キャビネットやハウジングの入口にシールドを施し、入口においても外乱を防止します。

2.9.2 通信バスケーブル

通信バスケーブルのシールドについては、関連するバスシステムの敷設説明書に記載されています。

2.9.3 信号線

アナログ信号用のバスモジュールおよび一部のインターフェースバスモジュールには、シールド用の接続端子が付いています。



メモ

表面積の大きな部分にあらかじめシールドを施しておくとしールド効果が高まります。これを実施するために、ワゴシールド結線システムの使用をお勧めします。

特に使用が推奨されるのは、システムの規模が大きく、一定電流が流れるか、ハイパルス電流（空中放電などによる）が発生するシステムです。

2.9.4 ワゴシールド（スクリーン）結線システム

ワゴシールド結線システムは、シールドクランプサドル、キャリア、および各種の取り付け部品で構成され、多様な構成を実現します。詳しくはカタログ W4-Vol.3 の 10 章を参照してください。



図 2-25 : ワゴシールド（スクリーン）結線システム

p0xxx08x、p0xxx09x、and p0xxx10x

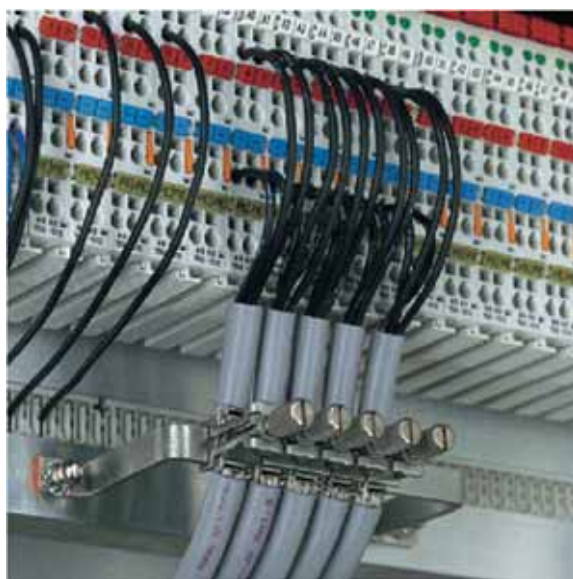


図 2-26 : ワゴシールド（スクリーン）結線システムの適用例

p0xxx11x

2.10 アセンブリのガイドラインおよび規格

DIN 60204	機械用電気装置
DIN/EN 50178	電子回路を備えた高電圧システムの装置（以前の VDE 0160 に対応）
EN 60439	低電圧開閉装置及び制御装置アセンブリ

3 フィールドバスコントローラ

3.1 フィールドバスコントローラ 750-837、750-838

3.1.1 概要

CANopen 用フィールドバスコントローラには、フィールドバス（CAN バス）への接続方法と内部メモリの違いによって以下のようなバリエーションがあります。

型番	フィールドバス接続方法
750-837 750-837/020-000 750-837/021-000	MCS（マルチコネクションシステム）コネクタ
750-838 750-838/020-000 750-838/021-000	D-Sub コネクタ（9 ピン）

型番	メモリ容量
750-837 750-837/020-000 750-837/021-000	PFCメモリ：RAM 64kB、FLASH 64/128kB*) PFCメモリ：RAM 192kB、FLASH 128/256kB*) PFCメモリ：RAM 832kB、FLASH 320/640kB*)
750-838 750-838/020-000 750-838/021-000	PFCメモリ：RAM 64kB、FLASH 64/128kB*) PFCメモリ：RAM 192kB、FLASH 128/256kB*) PFCメモリ：RAM 832kB、FLASH 320/640kB*)

*)オンライン変更あり/なし



メモ

本取扱説明書では関連する図や写真は 750-837 のものを掲げています。機能説明は 750-837/020-000、/021-000、750-838、/020-000、021-000 にも同様に適用できます。

プログラマブルフィールドバスコントローラ（PFC）は、フィールドバスカプラ（750-337、750-338 など）の CANopen 機能と PLC の機能とを組み合わせたものです。

アプリケーションプログラムは IEC61131-3 に基づいた WAGO-I/O-PRO CAA によって作成します。

センサの入力信号全てはコントローラでグループ化されます。

データ処理は IEC61131-3 プログラミングに従って、ローカルの PFC で行われます。この方法で作成したデータの結果はアクチュエータに直接出力するか、バス経由で上位の制御システムに送信することができます。

プロセスデータの交換は PDO および SDO を用いて行われます。コントローラは 32 個の TxPDO と 32 個の RxPDO、また CANopen フィールドバス経由でプロセスデータを送るための 2 個のサーバ SDO と 16 個のクライアント SDO をサポートします。

プログラマはフィールドバスデータと I/O データの全てにアクセスできます。

バスコントローラはイニシャライゼーションフェーズでノードの物理的構造を判断し、全ての入出力のプロセスイメージを生成します。このときアナログモジュール（ワード単位 of データ交換）およびデジタルモジュール（バイト単位 of データ交換）の混在があるかもしれません。

バスコントローラ内のプロセスイメージは、入力および出力領域に分けられます。アナログモジュールのデータは、バスコントローラからの配置順序に従って PDO にマッピングされます。デジタルモジュールのビット集合は、バイトの形に変換してから PDO にマッピングされます。デジタル I/O の数が 8 ビットを超えた場合、コントローラは自動的に次のバイトに

移ります。

オブジェクトディクショナリの入力項目は、必要に応じ 32 個の RxPDO および 32 個の TxPDO にマッピングすることができます。入出力データ領域全体は SDO で送信することができます。

「スペーサモジュール」はソフトウェアを用いて設定することができます。

3.1.2 適合性

本書記載のコントローラは以下の表のように、WAGO-I/O-PRO 32 および WAGO-I/O-PRO CAA プログラミングツールの各バージョンに対応しています。

プログラミング ツール コントローラ	WAGO-I/O-PRO 32 759-332		WAGO-I/O-PRO CAA 759-333				
	V2.1	V2.2.6	V2.3.6.2	V2.3.7.2	V2.3.8.5	V2.3.9.7	V2.3.9.13
750-837	✓	✓	HW≥06 SW≥11	HW≥06 SW≥11	HW≥06 SW≥11	HW≥06 SW≥11	HW≥06 SW≥11
750-837/020-000	—	—	✓	✓	✓	✓	✓
750-837/021-000	—	—	✓	✓	✓	✓	✓
750-838	—	—	✓	✓	✓	✓	✓
750-838/020-000	—	—	✓	✓	✓	✓	✓
750-838/021-000	—	—	✓	✓	✓	✓	✓

—	コントローラは当該 WAGO-I/O-PRO バージョンに適合しない
✓	コントローラはハードウェア、ソフトウェアのバージョンにかかわらず、当該 WAGO-I/O-PRO バージョンに適合する
HW≥06	コントローラはハードウェアバージョンが 06 およびそれ以上ならば、当該 WAGO-I/O-PRO バージョンに適合する
SW≥11	コントローラはソフトウェアバージョンが 11 およびそれ以上ならば、当該 WAGO-I/O-PRO バージョンに適合する



注意

ゲートウェイオブジェクトをサポートし、実行を保証するためには以下のバージョンのハードウェアおよびソフトウェア以降を使用してください。

— 750-837、/020-000、/021-000 = HW06、SW11

— 750-838、/020-000、/021-000 = HW01、SW11

SW11 以降のソフトウェアバージョンに関しては、上記に示されたハードウェアの条件は絶対に必要なものとなります。同様に、ハードウェアが規定されたバージョン以降であっても、ソフトウェアバージョン SW11 は最低限必要なものとなります。

HW06 以前のハードウェアバージョンの 750-837 コントローラは SW10 までのソフトウェアバージョンでのみ実行可能ですが、ゲートウェイオブジェクトはサポートしていません。

3.1.3 ハードウェア

3.1.3.1 外観

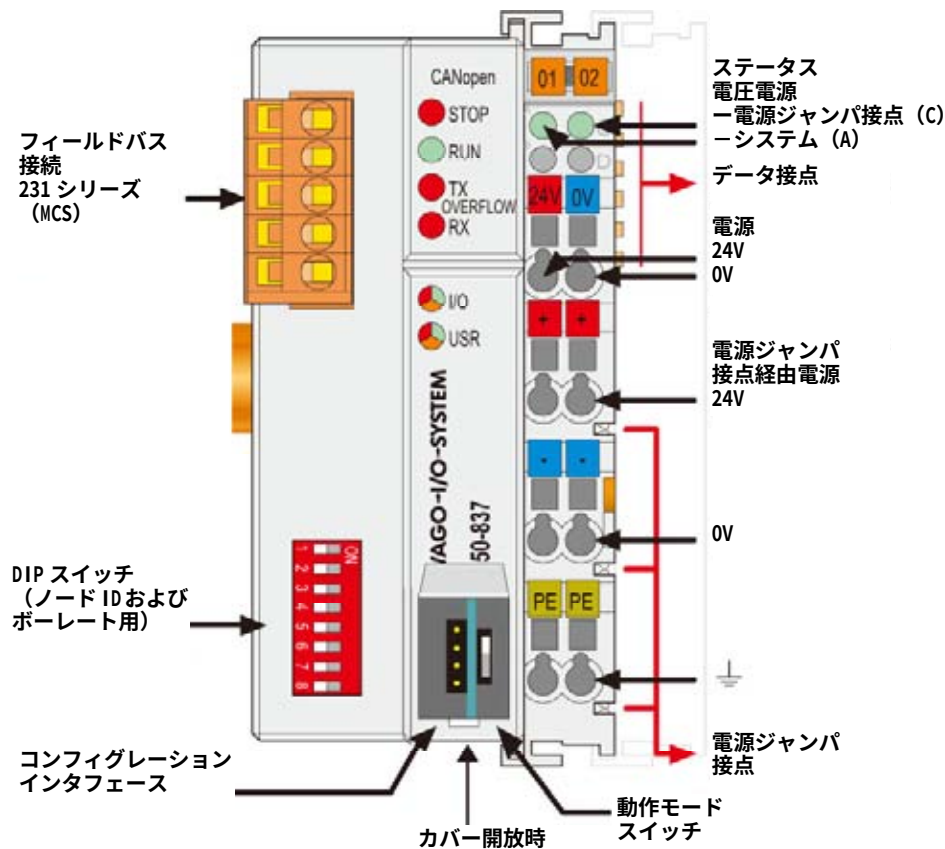


図 3-1 : CANopen フィールドバスコントローラ 750-837

g083700e

このフィールドバスコントローラは、以下の部分により構成されます。

- 電源モジュール（システム給電を行う内部システム電源モジュール、および I/O モジュールアセンブリを介してフィールド給電を行う電源ジャンパ接点を装備）
- 通信バス接続によるフィールドバスインタフェース
- ボーレートとノード ID を設定する DIP スイッチ
- 表示ランプ（LED）：動作状態、バス通信、動作電圧、および故障メッセージと診断結果を表示
- コンフィグレーション/プログラミングインタフェースおよび動作モードスイッチ
- I/O モジュール（内部バス）およびフィールドバスインタフェースとの通信を行う電子回路部

3.1.3.2 種類

対応するバスコントローラ製品は以下のとおりです。

型番	内容
750-837	CANopen コントローラ、MCS コネクタ接続 PFC メモリ : RAM 64kB、FLASH 64/128kB (オンライン変更あり/なし)
750-837/020-000	CANopen コントローラ、MCS コネクタ接続 PFC メモリ : RAM 192kB、FLASH 128/256kB (オンライン変更あり/なし)
750-837/021-000	CANopen コントローラ、MCS コネクタ接続 PFC メモリ : RAM 832kB、FLASH 320/640kB (オンライン変更あり/なし)
750-838	CANopen コントローラ、D-SUB コネクタ接続 PFC メモリ : RAM 64kB、FLASH 64/128kB (オンライン変更あり/なし)
750-838/020-000	CANopen コントローラ、D-SUB コネクタ接続 PFC メモリ : RAM 192kB、FLASH 128/256kB (オンライン変更あり/なし)
750-838/021-000	CANopen コントローラ、D-SUB コネクタ接続 PFC メモリ : RAM 832kB、FLASH 320/640kB (オンライン変更あり/なし)

3.1.3.3 デバイス電源

電源はケージクランプ®を装備した端子台に給電します。デバイス電源部は、システムユニットとフィールドユニットに対して電源を供給します。

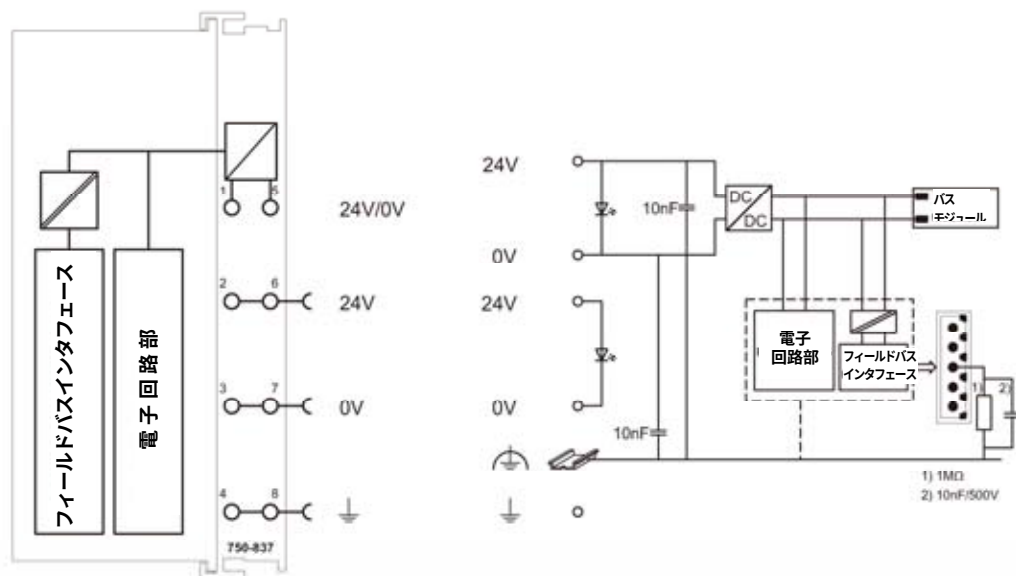


図 3-2 : デバイス電源

g083701e

内部に組み込まれたシステム電源回路は、バスコントローラの電子回路部と各 I/O モジュールに必要な電圧を生成します。

フィールドバスインタフェース部は、内部システム電源回路から絶縁した電圧で給電されます。

3.1.3.4 フィールドバス接続

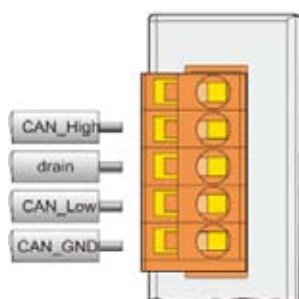
750-837、 /020-000、 /021-000	フィールドバスコントローラ 750-837、/020-000、/021-000 は、CAN バスへ接続するためにオープン型コネクタ仕様となっています。 MCS (マルチコネクションシステム、231 シリーズ) コネクタ標準装備
750-838、 /020-000、 /021-000	フィールドバスコントローラ 750-838、/020-000、/021-000 は、CAN バスへ接続するために 9 ピン D-SUB オスコネクタを装備しています。 (バスコネクタ 750-963 は含まれていませんので別途お求めください)

通信コネクタが挿入される場所の高さは低めになっており、80mm 高のボックス内での取り付けも可能になっています。

フィールドバスシステムと電子回路間は、DC/DC コンバータとフィールドバスインタフェース部のオプトカプラにより、電氣的に絶縁がされています。

上記両方のフィールドバスコントローラ共、CANopen 通信ケーブルのシールドはドレイン端子に接続しなければなりません。シールドは 1MΩ 抵抗を介してグラウンド（アース）（レールキャリア接点）に接続されます。シールドとグラウンド（アース）間を低インピーダンスで接続するのは外部（電源端子台など）でのみ実現できます。CAN バス全体のシールドについては、グラウンド（アース）を中央で取ることを推奨します。

3.1.3.4.1 MCS (750-837、/020-000、/021-000)

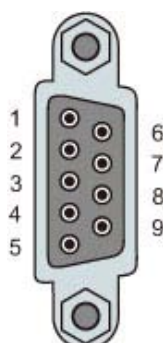


ピン	信号	説明
5	CAN_V+	不使用
4	CAN_H	CAN 信号 High
3	ドレインシールド	シールド接続
2	CAN_L	CAN 信号 Low
1	GND	グラウンド

図 3-3 : 750-837 フィールドバス接続用 231 (MCS) シリーズコネクタピン配置

g012400x

3.1.3.4.2 D-SUB (750-838、/020-000、/021-000)



ピン	信号	説明
1	—	不使用
2	CAN_L	CAN 信号 Low
3	GND	グラウンド
4	—	不使用
5	ドレインシールド	シールド接続
6	—	不使用
7	CAN_H	CAN 信号 High
8	—	不使用
9	CAN_V+	不使用

図 3-4 750-838 フィールドバス接続用 D-SUB コネクタピン配置

g012448x

3.1.3.5 表示素子

フィールドバスコントローラまたはノードの動作状態は LED によって表示されます。

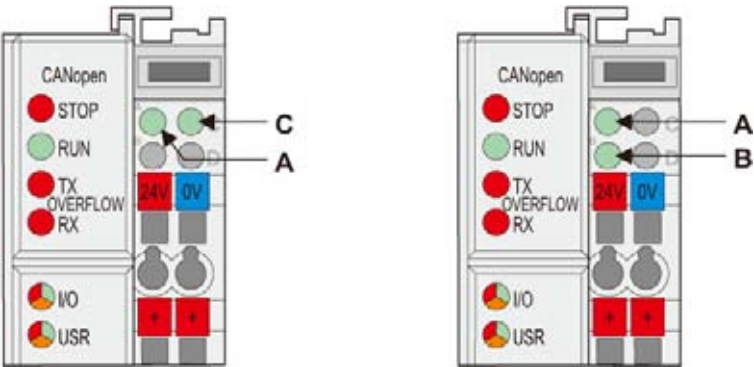


図 3-5 : CANopen 表示素子

g012455x

LED	色	意 味
STOP	赤	バスコントローラ及びノードは STOP ステート
RUN	緑	バスコントローラ及びノードは OPERATIONAL ステート
Tx-Overflow	赤	CAN 送信バッファがフル
Rx-Overflow	赤	CAN 受信バッファがフル
I/O	赤/緑/橙	ノードの動作やエラー信号を表示
USR	赤/緑/橙	コントローラのユーザプログラムによって選択できる
A	緑	システム電源表示
B または C	緑	フィールド電源表示（製造時期により異なります）



詳細情報

LED の表示内容の詳細に関しては 3.1.9 節「LED 表示」を参照してください。

3.1.3.6 コンフィグレーション/プログラミングインタフェース

コンフィグレーション/プログラミングインタフェースは WAGO-I/O-CHECK との通信またはファームウェアダウンロード用にもちいられ、使用するときにはインタフェース用カバーを開けます。

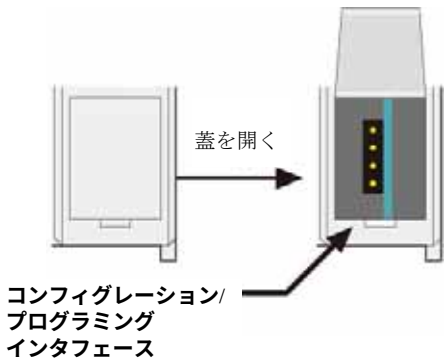


図 3-6 : コンフィグレーション/プログラミングインタフェース

g01xx06e

PC とつなぐためには、専用の通信ケーブル（750-920）を上図の 4 極ヘッダに接続します。



注意

通信ケーブル 750-920 は、コントローラの電源を入れたまま抜き差ししないようにしてください。

3.1.3.7 動作モードスイッチ

動作モードスイッチはインタフェース用カバーを開けたとき、コンフィグレーション/プログラミングインタフェースの横にあります。

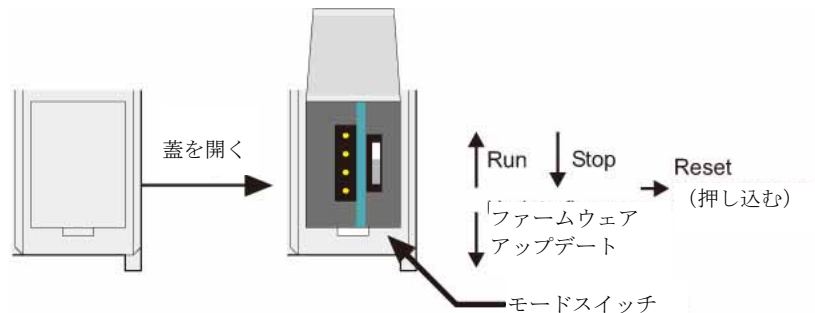


図 3-7 : 動作モードスイッチ

モードスイッチは押し下げ/スライド型スイッチで、設定位置が 3 箇所あり、RUN—Stop 切り替えの機能を持っています。

動作モードスイッチ位置	機 能
中央から上端へ	プログラム処理の実行 (RUN)
上端から中央へ	プログラム処理の停止 (STOP)
下端、ブートストラップ	ファームウェアのロード用、ユーザは使用不可です。
押し下げ (ドライバを使用)	ハードウェアリセット 全ての出力がリセットされます。変数は 0、FALSE、初期値のいずれかにセットされます。 リティン変数またはフラグは変更されません。 ハードウェアリセットは、STOP 位置または動作モードスイッチの位置にかかわらず RUN 状態で実行することができます。



注意

動作モードスイッチを RUN から STOP に切り替えたとき出力がセットされていた場合、その出力はセットのままとなります。ソフトウェアツールで出力を OFF にしようとしても無効です。何故ならばプログラムがもう処理されないからです。



メモ

WAGO-I/O-PRO CAA の中で"GET_STOP_VALUE" ("System.lib"ライブラリ) を用いると、プログラムストップ前の最終サイクルを認識する機能が提供されますので、ユーザは STOP が起きたときのためにコントローラのビヘイビアをプログラムすることができます。この機能を利用すれば、コントローラの出力を安全状態に切り替えることができます。

3.1.3.8 ハードウェアアドレス (モジュール ID)

DIP スイッチは、フィールドバスコントローラの通信速度およびモジュール ID の設定の両方に用いられます。このモジュール ID は COB ID の計算 (PDO1~4、サーバ SDO など) に必要となります。

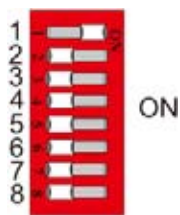


図 3-8 : 局 (ノード) アドレス (ID1 の例)

g012440x

個々の DIP スイッチの位置はバイナリ数に対応しており、スイッチ番号に応じて増えます。例えばモジュール ID1 は DIP1=ON で、モジュール ID8 は DIP4=ON でセットされます。

ワゴ I/O システムのノードは、モジュール ID が 1～127 の範囲となります。

3.1.3.9 通信速度の設定

このバスコントローラは 9 種類の通信速度 (ボーレート) を設定できます。ボーレートの設定は DIP スイッチによって行います。

モジュール ID=0 (全ての DIP スイッチを OFF) をセットし、電源を ON するとバスコントローラはコンフィグレーションモードになります。ここで現在設定済みのボーレートが表示されます。ボーレートは上部 4 個の LED (STOP、RUN、Tx-、Rx-Overflow) で表示されます。ここで、LED と各スイッチの対応は、STOP=SW1、RUN=SW2、Tx-Overflow=SW3、Rx-Overflow=SW4 となります。

現在のボーレートは対応する LED の遅い点滅で分かります。新しいボーレートは対応するスイッチを ON にすることで設定されます。

設定したコンフィグレーションは DIP8 を ON にしてセーブします。セーブした後、新しいボーレートは対応 LED が一定に点灯することで確認できます。1 MBaud のボーレートのときのみ 4 個の LED 全てが点滅をします。

例 : 125kB=Tx-Overflow LED が点滅
250kB=STOP および RUN LED が点滅



図 3-9 : 例 : ボーレート 125kB のセーブ

g012441x

この段階では、CAN 経由でのデータ交換はできません。

DIP SW 番号	機能	1M baud	800 kB	500 kB	250 kB	125 kB	100 kB	50 kB	20 kB	10 kB	LED 表示
1(LSB)	ボーレート	0	1	0	1	0	1	0	1	0	STOP
2	ボーレート	0	0	1	1	0	0	1	1	0	RUN
3	ボーレート	0	0	0	0	1	1	1	1	0	Tx-Overflow
4(MSB)	ボーレート	0	0	0	0	0	0	0	0	1	Rx-Overflow
5											
6											
7											
8	決定	オフ→オンでコンフィグレーション設定を決定									

ボーレート設定または確認が終了したとき電源を OFF にしてください。電源 OFF の間に DIP スイッチによってモジュール ID をセットします。次に電源を ON した時、DIP スイッチで設定した値により ID がセットされます。
ボーレートのデフォルト値は 125kB です。

3.1.4 オペレーティングシステム

3.1.4.1 スタートアップ

コントローラは電源を入れた後、またはハードウェアリセットを行った後にスタートアップします。このとき、フラッシュメモリ内の PLC プログラムは RAM に転送されます。

このスタートアップに続いて、システムのイニシャライゼーションが行われます。コントローラは各 I/O モジュールと現在のコンフィグレーションを決定します。変数は 0、FALSE または PLC プログラムによって与えられた初期値にセットされます。フラグはその状態を維持します。このフェーズの間”I/O”LED は赤色で点滅をします。

スタートアップ時エラーがなければ、その後コントローラは”RUN”モードに移行します。”I/O”LED は緑色に点灯します。



詳細情報

LED の表示内容の詳細に関しては 3.1.9 節「LED 表示」を参照してください。

製品の出荷時は、PLC プログラムはまだフラッシュメモリに存在しません。この場合、コントローラはシステムをイニシャライズしないで立ち上がります。従ってこのときは、バスカプラとして動作します。

3.1.4.2 PLC サイクル

PLC サイクルは、動作モードスイッチが上端の位置にあるとき、または WAGO-I/O-PRO CAA からスタートコマンドを受けたとき、スタートアップ時エラーがなければ開始します。フィールドバスと I/O モジュールの入出力データ、および時間データが読み出されます。続いて RAM 内の PLC プログラムが処理され、フィールドバスやプロセスイメージ内の I/O モジュールの出力データが書き込まれます。オペレーティングシステム機能の中で、特に診断や通信などが実行され、PLC サイクルの最後に時間が更新されます。このサイクルは、入出力データや時間の読み出しをすることで再び開始します。

動作モード (STOP/RUN) の変化は PLC サイクルの最後で行われます。

サイクルタイムは PLC プログラムの開始から次の開始までの時間をいいます。PLC プログラム内でループ処理がプログラムされた場合、PLC 実行時間すなわち PLC サイクルタイムは、それにより延長されます。

入力、出力、時間は PLC プログラムの処理中に変更されません。この変更は、PLC プログラムの最後でのみ定義した方法で行われます。このためプロセスからのイベントや、ループ内の時間経過を待つことはできません。

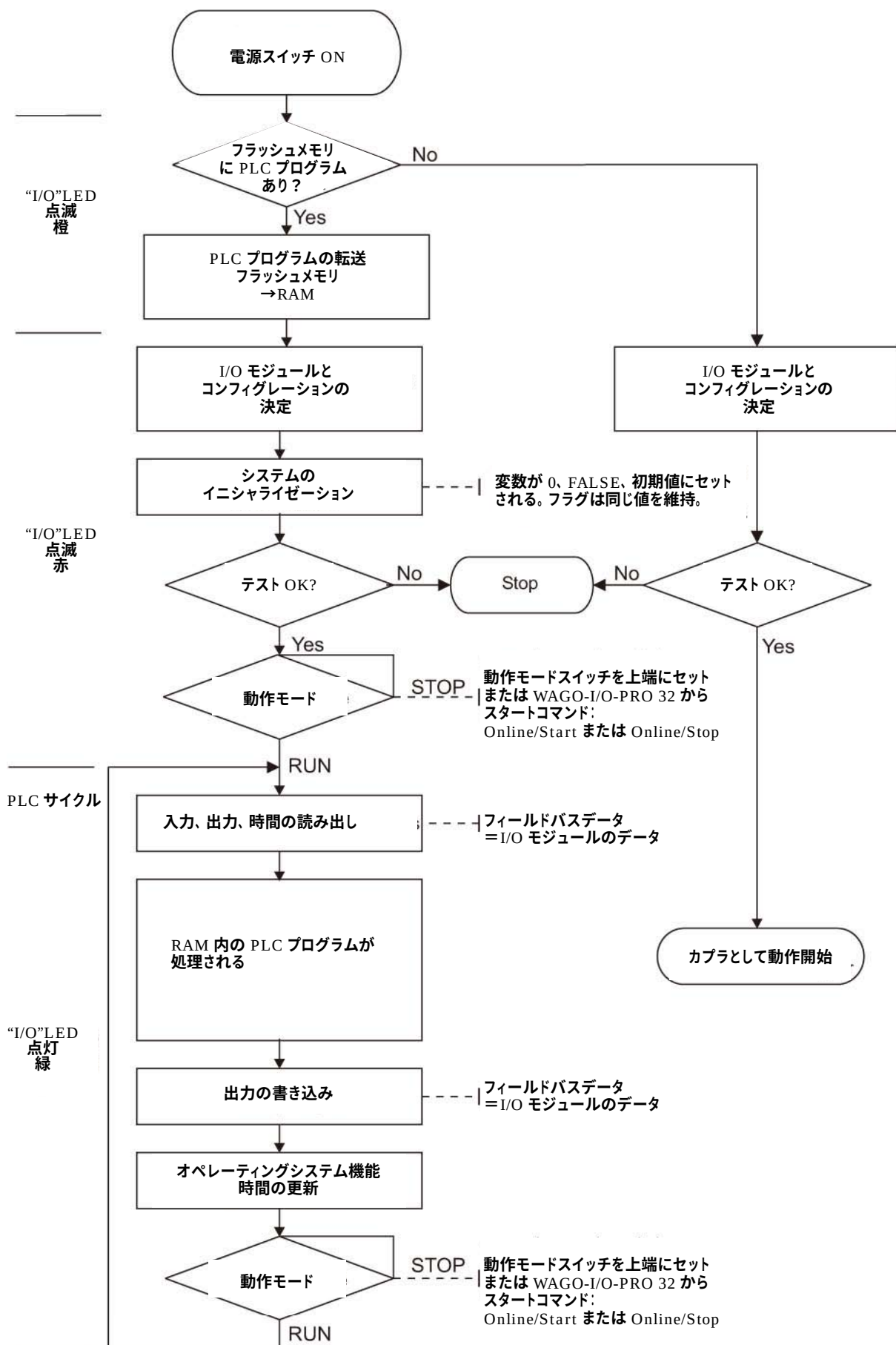


図 3-10 : コントローラオペレーティングシステム

g012x41x

3.1.5 プロセスイメージ

3.1.5.1 データ構成の原則

電源を投入した後、コントローラはデータ（データ幅/ビット幅>0）の入出力用にノードに挿入した全ての I/O モジュールを認識します。ノード内では、アナログおよびデジタル I/O は混在することができます。

コントローラはデータ幅や I/O モジュールのタイプ、またノード内の I/O モジュールの位置などを基にして、内部にプロセスイメージを作成します。このプロセスイメージは、入力および出力データ領域に分けられます。

デジタル I/O モジュールのデータはビット単位であり、データ交換はビット毎に行われます。アナログ I/O モジュールは全てバイト単位であり、データ交換はバイト毎に行われます。アナログ I/O モジュールには、例えばカウンタモジュール、位置測定モジュール、通信モジュールなどがあります。



メモ

I/O モジュールの入出力ビット数またはバイト数については、各 I/O モジュールの説明を参照してください。

I/O モジュールのデータは入力と出力のローカルプロセスイメージに分割され、各プロセスイメージ内ではコントローラの後に置いた順序で配置されます。

それぞれの I/O 領域では、初めにアナログ I/O モジュールのマッピングが行われた後、デジタル I/O モジュールのマッピングが行われます。実際に接続されたアナログおよびデジタルの I/O モジュールの順番が逆の場合であっても、この順番で行います。デジタルチャンネルは、各グループのデータ幅が 1 バイトとなるようにグループ化します。デジタル I/O の数が 8 ビットを超えると、コントローラは次のバイトに自動的に移ります。



注意

ノードを変更または拡張すると、プロセスイメージの構成が変わる場合があります。その場合、プロセスデータのアドレスも以前のアドレスから変更されます。モジュールを追加したときは、それまでに搭載していた全モジュールのプロセスデータを考慮に入れてください。

物理的なモジュールのデータ用プロセスイメージは、ワゴ CANopen バスカブラの場合と同じです。コントローラに関しては、PFC 変数のデータがプロセスイメージの中で入力および出力データに基づいて分けられ、物理データの後に加えられます。



詳細情報

ワゴ I/O システム 750 および 753 の全ての I/O モジュールに対して、フィールドバス固有のプロセスデータ構造に関しては「CANopen 用プロセスデータ構造」の章を参照してください。

3.1.6 データ交換

CANopen においては、データ送信、イベントのトリガ、エラー状態の通知などは通信オブジェクトを用いて行われます。各通信オブジェクトは最大 8 バイトのデータを含んだ CAN テレグラムで構成されます。このテレグラムはネットワーク内で固有の COB-ID（通信オブジェクト ID）が割り当てられます。

通信オブジェクトのパラメータおよび CANopen サブスクライバのパラメータやデータは、オブジェクトディクショナリに載っています。

3.1.6.1 通信オブジェクト

バスコントローラ 750-837、/020-000、/021-000 および 750-838、/020-000、/021-000 は以下の通信オブジェクトをサポートします。

- 32×Tx-PDO
フィールドバスノード入力データのプロセスデータ交換用
- 32×Rx-PDO
フィールドバスノード出力データのプロセスデータ交換用
- 2×サーバ SDO
コンフィグレーションデータの交換およびノードの状態についての通知用
- 16×クライアント SDO
自己ノードまたは他の 16 台のフィールドバスノードに対するコンフィグレーションデータの要求用
- 同期オブジェクト（SYNC）
ネットワーク同期用
- エマージェンシオブジェクト（EMCY）
- ネットワーク管理オブジェクト
 - ーモジュール制御プロトコル
 - ーエラー制御プロトコル
 - ーブートアッププロトコル

3.1.6.2 通信インタフェース

CANopen バスコントローラは、データ交換に関し 3 種類のインタフェースを持っています。

- フィールドバス（マスタ）へのインタフェース
- PFC（CPU）による PLC 機能
- バスモジュールへのインタフェース

データ交換はフィールドバスマスタとバスモジュールの間、PFC（CPU）による PLC 機能とバスモジュール間、そしてフィールドバス（マスタ）と PFC（CPU）による PLC 機能間で行われます。

3.1.6.3 メモリ領域

コントローラは物理的入出力データ用に 256 ワード（ワード 0～255）のメモリ空間を使用します。

コントローラには、IEC61131-3 に従って定義された PFC 変数をマッピングするための追加メモリ空間が割り当てられています。

この拡張メモリ空間（ワード 256～511：各入出力）は、物理プロセスイメージの後に PFC 変数をマッピングするのに用いられます。

メモリ空間の分割とプロセスデータへの PLC 機能（CPU）のアクセスの仕方は、ワゴの全てのフィールドバスコントローラと同じです。アクセスはアプリケーションに関連した IEC61131-3 プログラム経由で行われ、フィールドバスシステムからは独立しています。

フィールドバス側からのアクセスはフィールドバスに固有となります。

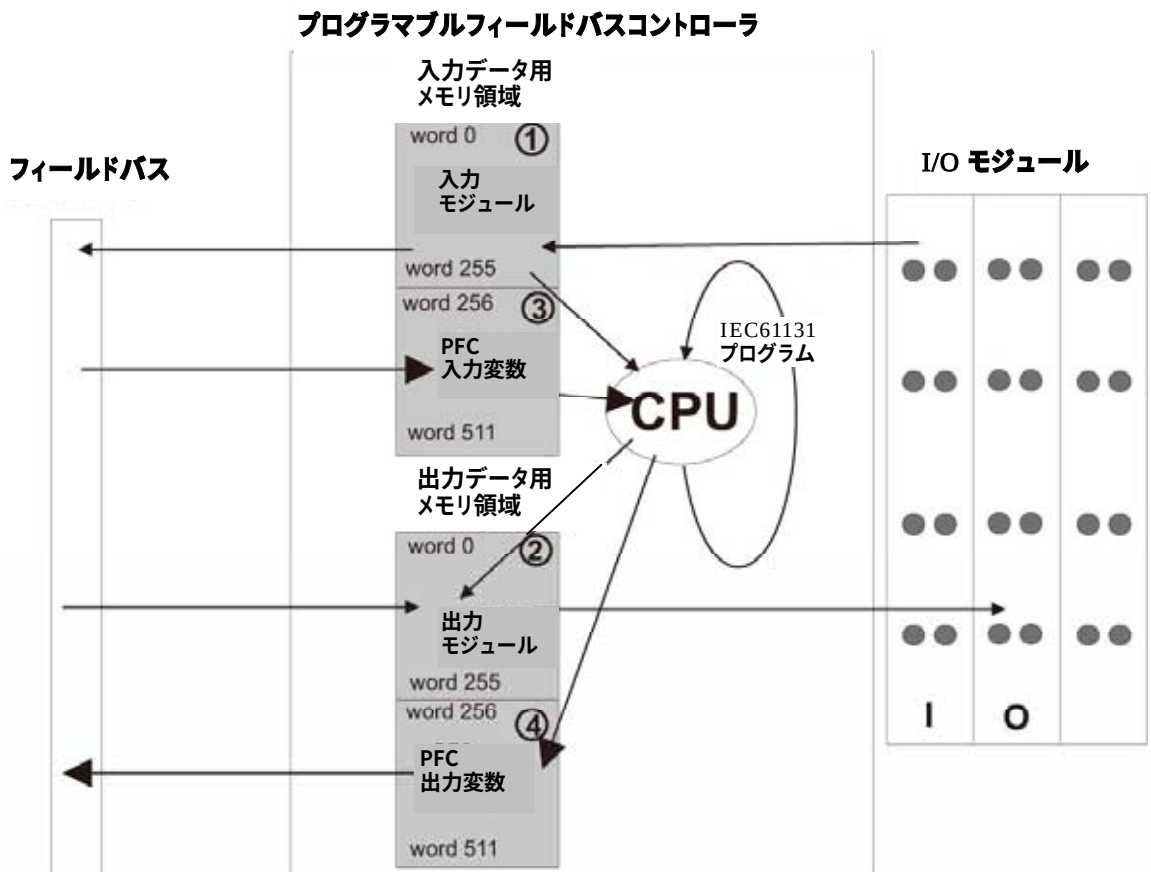


図 3-11：バスコントローラのメモリ領域およびデータ交換

g012434e

コントローラのプロセスイメージにおいてワード 0～255 のメモリ空間には、バスモジュールの物理データを含みます。

- ① 入力モジュールデータは CPU およびフィールドバス側から読むことができます。
- ② 同様に出力モジュールへの書き込みは、CPU およびフィールドバス側から行うことができます。

PFC 変数はプロセスイメージ内ワード 256～511 のメモリ空間に置かれます。

- ③ PFC 入力変数はフィールドバス側から入力メモリ空間に書き込まれ、その後の処理のために CPU によって読み取られます。
- ④ IEC61131-3 プログラムを介して CPU によって処理される変数は出力メモリ空間に置かれ、マスタによって読み取ることができます。

コントローラは上記以外にもメモリ空間を保有していますが、中にはフィールドバス側からアクセスできない領域もあります。

RAM メモリ	インタフェースとの通信ではなく内部処理（結果の計算など）に必要とされる変数の生成に使用されます。
リテインメモリ	リテインメモリは不揮発性メモリです。すなわち電源が故障しても全ての値が保持されます。メモリ管理は自動的に行われます。IEC61131-3 プログラム用にこのメモリ領域で使われるフラグは、メモリ空間アドレス指定のない変数や「var retain」で明示的に変数定義された変数と共に保存されます。
	メモ メモリ管理が自動的に行われることにより、データのオーバーラップを引き起こす可能性があります。従って、フラグとリテイン変数は混合して使わないことをお勧めいたします。
プログラムメモリ	プログラムメモリには IEC 61131-3 プログラムが格納されます。このプログラムメモリはフラッシュメモリです。電源を投入するとプログラムがフラッシュメモリから RAM に転送されます。コントローラが問題なく立ち上がったとき、動作モードスイッチが最上位置にあるか、または WAGO-I/O-PRO CAA から起動コマンドが送られた場合、PFC サイクルが開始されます。

3.1.6.4 アドレッシング

3.1.6.4.1 フィールドバス固有アドレス指定

電源を投入したとき、データはプロセスイメージからオブジェクトディクショナリにマッピングされます（イニシャライゼーション）。CANopen フィールドバスマスタは、PDO または SDO 経由でデータをアドレス指定してアクセスするために、オブジェクトディクショナリの 16 ビットのインデックスと 8 ビットのサブインデックスを用います。

従って、プロセスイメージ内のデータの位置は、CANopen ユーザにとって直接意味がありません。

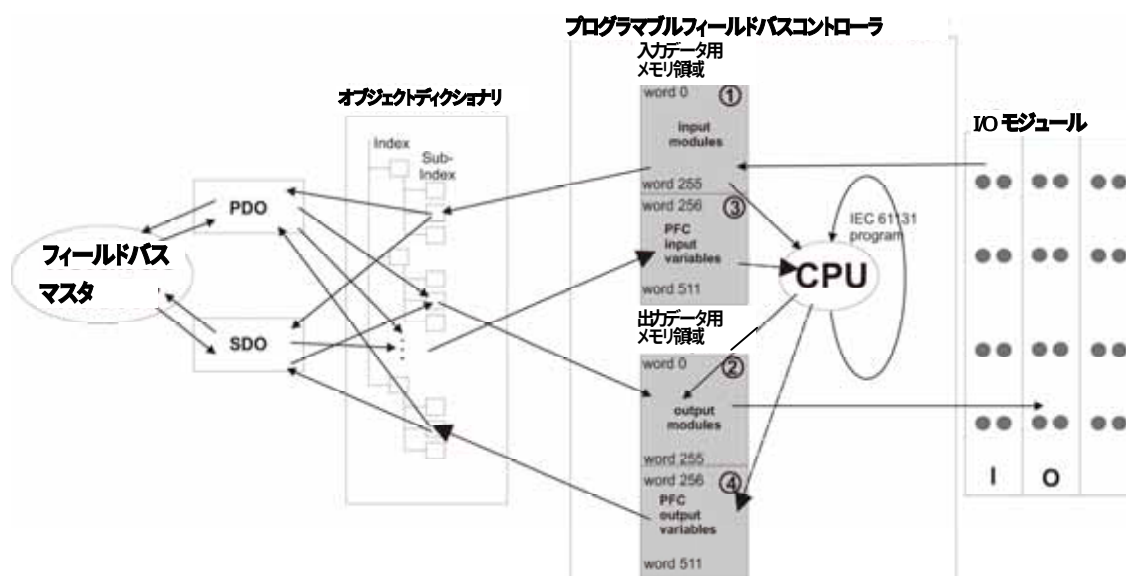


図 3-12 : CANopen バスコントローラ用フィールドバス固有データ交換

g012432e

3.1.6.4.1.1 I/O モジュールデータのインデックス

ユーザ固有のコンフィグレーションがイニシャライゼーション前に保存され、また現在接続されたモジュールのコンフィグレーションが以前保存されたコンフィグレーションと一致した場合、イニシャライゼーションはそのコンフィグレーションで行われます。



メモ

ユーザ固有のコンフィグレーションに対するイニシャライゼーションの例に関しては、3.1.8.6 節「アプリケーション固有のマッピング」を参照してください。

他の場合は全て、イニシャライズするときデバイスプロファイル DS401 に基づいて、デフォルトのコンフィグレーションがオブジェクトディクショナリに割り当てられます。

ここで、オブジェクトディクショナリへの登録はデータ幅（1 ビット、1 バイト、2 バイト、3 バイト、その他）や入力・出力によって別々に行われます。

バスモジュールの物理的な配置はノード内のどこでも可能です。

データ幅=1 ワード/チャンネル	データ幅=1 ビット/チャンネル
アナログ入力モジュール	デジタル入力モジュール
アナログ出力モジュール	デジタル出力モジュール
熱電対入力モジュール	診断付デジタル出力モジュール
抵抗センサ入力モジュール	電源入力モジュール（ヒューズ/診断付あり）
パルス幅出力モジュール	ソリッドステートリレー
インタフェースモジュール	リレー出力モジュール
アップダウンカウンタ	

表 3-1 : I/O モジュール データ幅



詳細情報

各 I/O モジュールの入出力ビット数またはバイト数に関しては、対応する I/O モジュールの記述を参照してください。

デジタルモジュールのデータが最初に考慮されます。

CANopen ではデータはビット単位で送信しないため、デジタルモジュールのデータはバイト単位でグループ化され、対応するインデックスに割り当てられます。すなわち、デジタル入力データはインデックス 0x2000（または 0x6000）、デジタル出力モジュールはインデックス 0x2100（または 0x6200）になります。

アナログモジュールなどの 1 バイト以上のデータ幅を持ったバスモジュールは、その幅に応じて各々のインデックスが割り当てられます。

以下の表はバスモジュールデータのインデックスを示しています。

データ巾	入力モジュール	出力モジュール
	インデックス	
1 bit デジタル	0x2000 (0x6000)	0x2100 (0x6200)
1 byte 特別モジュール	0x2200	0x2300
2 byte 特別モジュール	0x2400 (0x6401)	0x2500 (0x6411)
3 byte 特別モジュール	0x2600	0x2700
4 byte 特別モジュール	0x2800	0x2900
5 byte 特別モジュール	0x3000	0x3100
6 byte 特別モジュール	0x3200	0x3300
7 byte 特別モジュール	0x3400	0x3500
8 byte 特別モジュール	0x3600	0x3700

表 3-2 : バスモジュールデータのオブジェクトディクショナリ内インデックス

各インデックスは最大 256 のサブインデックス (0~255) を持ちます。
サブインデックス 0 には総データブロック数 (バイト数) が入ります。サブインデックス 1
以降にはブロック毎にデータが入力されます。ブロックサイズはバスモジュールのデータ幅
によって異なります。

サブインデックス	内 容
0	データブロック数
1	1 番目データブロック (I/O モジュールのデータ幅)
2	2 番目データブロック (I/O モジュールのデータ幅)
...	...

表 3-3 : バスモジュールデータのオブジェクトディクショナリ内サブインデックス



メモ

デフォルトコンフィグレーションの設定の詳細に関しては、4.3.4.1 節「イニシャライゼーション」参照してください。



注意

ノードが変更または拡張された場合、プロセスイメージを再構成することになります。この
場合、プロセスデータのアドレスもまた前回のものと比較して変更になります。モジュール
を追加した場合、全ての既存モジュールのプロセスデータを考慮に入れてください。

例 :

バスモジュールのコンフィグレーションが以下の場合

- 1) 2 チャンネルデジタル入力モジュール×5 (750-400 など)
- 2) 4 チャンネルデジタル出力モジュール×1 (750-504 など)
- 3) 2 チャンネルアナログ出力モジュール×2 (750-552 など)

1) 2ch デジタル入力モジュール×5 データのインデックス

インデックス	サブインデックス	内容	説明
0x2000 (0x6000)	0	2	8 ビット入力ブロック数
	1	D4. 2 D4. 1 D3. 2 D3. 1 D2. 2 D2. 1 D1. 2 D1. 1*	デジタル入力ブロック 1
	2	0 0 0 0 0 0 D5. 2 D5. 1	デジタル入力ブロック 2

*D1. 1=ビットデータモジュール 1、チャンネル 1、D1. 2=ビットデータモジュール 1、チャンネル 2、他

2) 4ch デジタル出力モジュール×1 データのインデックス

インデックス	サブインデックス	内容	説明
0x2100 (0x6200)	0	1	8 ビット出力ブロック数
	1	0 0 0 0 D1. 4 D1. 3 D1. 2 D1. 1*	デジタル出力ブロック

*D1. 1=ビットデータモジュール 1、チャンネル 1、D1. 2=ビットデータモジュール 1、チャンネル 2、他

3) 2ch アナログ出力モジュール×2 データのインデックス

インデックス	サブインデックス	内容	説明
0x2500 (0x6411)	0	4	2 バイトチャンネル数
	1	D1. 1*	アナログチャンネル 1
	2	D1. 2*	アナログチャンネル 2
	3	D2. 1*	アナログチャンネル 3
	4	D2. 2*	アナログチャンネル 4

*D1. 1=ワードデータモジュール 1、チャンネル 1、D1. 2=ワードデータモジュール 1、チャンネル 2、他

3.1.6.4.1.2 PFC 変数のインデックス

オブジェクトディクショナリへの PFC 変数の登録はデータタイプ (Integer8、Unsigned8、
Boolean、Integer16、その他) や入力・出力によって別々に行われます。

CANopen はデータをビット単位で送信しないという事により、Boolean データタイプの変
数データは、バイトにグループ化して対応するインデックスに割り当てられます。

Boolean タイプの入力変数データはインデックス 0xA080 に、Boolean タイプの出力変数データはインデックス 0xA500 に割り当てられます。変数データの割り当てが 1 バイト以上のデータ幅のものはアナログモジュールであり、個別のインデックスに割り当てられます。



メモ

IEC61131-3 入力変数は CAN バスの観点から定義されており、PFC では出力変数とみなされます。従って、IEC61131-3 出力変数は PFC では入力変数とみなされます。

データタイプ	IEC61131-3 入力変数	IEC61131-3 出力変数
	インデックス	
Integer8	0xA000	0xA480
Unsigned8	0xA040	0xA4C0
Boolean	0xA080	0xA500
Integer16	0xA0C0	0xA540
Unsigned16	0xA100	0xA580
Integer24	0xA140	0xA5C0
Unsigned24	0xA180	0xA600
Integer32	0xA1C0	0xA640
Unsigned32	0xA200	0xA680
Float32	0xA240	0xA6C0
Unsigned40	0xA280	0xA700
Integer40	0xA2C0	0xA740
Unsigned48	0xA300	0xA780
Integer48	0xA340	0xA7C0
Unsigned56	0xA380	0xA800
Integer56	0xA3C0	0xA840
Integer64	0xA400	0xA880
Unsigned64	0xA440	0xA8C0

表 3-4 : IEC61131-3 変数データのオブジェクトディクショナリ内インデックス

全てのインデックスは最大 256 (サブインデックス 0~255) のサブインデックスを持っています。

入力データの数はサブインデックス 0 で示されます。一方、各データは後続のサブインデックスにブロック単位で書き込まれます。

ブロックサイズは特定のデータタイプのデータ幅によって異なります。

サブインデックス	内 容
0	データブロック数
1	1 番目データブロック (データタイプに応じたデータ幅)
2	2 番目データブロック (データタイプに応じたデータ幅)
...	...

表 3-5 : IEC61131-3 変数データのオブジェクトディクショナリ内サブインデックス



メモ

デフォルトコンフィグレーションの設定の詳細に関しては、4.3.4.1 節「イニシャライゼーション」参照してください。

例 :

IEC61131-3 変数が以下のように定義された場合

- 1) Boolean データタイプの入力変数×11
- 2) Integer24 データタイプの出力変数×5

1) Boolean 入力変数×11 データのインデックス

インデックス	サブ インデックス	内 容	説 明
0xA080	0	2	8 ビット Boolean 入力ブロック数
	1	D8 D7 D6 D5 D4 D3 D2 D1*	Boolean 入力ブロック 1
	2	0 0 0 0 0 D11 D10 D9*	Boolean 入力ブロック 2

*D1=ビットデータ：入力変数 1、D2=ビットデータ：入力変数 2、他

2) Integer24 タイプ出力変数×5 データのインデックス

インデックス	サブ インデックス	内 容	説 明
0xA5C0	0	5	3 バイト出力ブロック数
	1	D1*	出力ブロック 1
	2	D2*	出力ブロック 2
	3	D3*	出力ブロック 3
	4	D4*	出力ブロック 4
	5	D5*	出力ブロック 5

*D1=3 バイトデータ：出力変数 1、D2=3 バイトデータ：出力変数 2、他

3.1.6.4.1.3 フィールドバスから PFC 変数データへのアクセス

プログラムは、1 バイトのデータ幅を持ったデータタイプ (Integer8、Unsigned8、Boolean) 用の特定インデックスを用いて、コントローラメモリに入っているデータへバイト単位でアクセスを許可します。各データに対する特定バイトはサブインデックスで選択します。大きなデータブロックに対しインデックスを用いたとき、数バイトへの同時アクセスが可能になります。

例えば Integer16 (0xA0C0) または Unsigned16 (0xA100) 用のインデックスは、書き込まれた PFC 出力変数データをワード単位で、また Integer24 用のインデックス 0xA140 は 3 バイト単位でアクセスすることを許可します。

例：

フィールドバスから PFC 出力データのバイト 512、バイト 513、バイト 514 へのアクセスは、以下のように Integer または Unsigned データタイプで行われます。

データアクセス	PFC 出力データ	インデックスでリードデータ (Integer/Unsigned)	サブインデ ックス
バイト単位 (Integer8/Unsigned8)	バイト 512	(0xA000 / 0xA040)	1
	バイト 513	(0xA000 / 0xA040)	2
	バイト 514	(0xA000 / 0xA040)	3
ワード単位 (Integer16/Unsigned16)	ワード 256 (バイト 512/513)	(0xA0C0 / 0xA100)	1
	ワード 257 (バイト 514/515)	(0xA0C0 / 0xA100)	2
3 バイト (Integer24/Unsigned24)	バイト 512～514	(0xA140 / 0xA180)	1

次ページ以降の表は、異なったデータ幅を持ったデータをアドレス指定するための考え方を示しています。

このために、フィールドバスコントローラ内の変数データ用に予約したメモリ空間は、データ幅に応じて個別にインデックスで割り当てられます。

表で示されたインデックス方法は、各々最大のインデックスおよびサブインデックスに達するまで続けられます。



メモ

プログラマブルフィールドバスコントローラの立場から見て、PFC 出力変数を定義するとします。一方で、CAN フィールドバスの立場から見ると、それらは入力変数とみなされます。同様に PFC 入力変数はフィールドバスがアクセスする IEC61131-3 出力変数とみなされます。

すなわち次のような対応となります：IEC61131-3 入力変数＝PFC 出力変数

PFC 入力変数＝IEC61131-3 出力変数

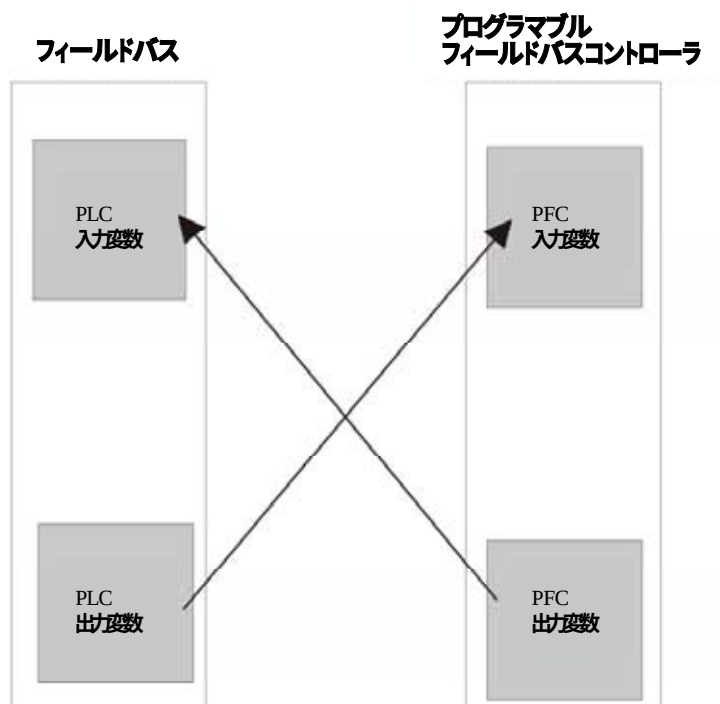


図 3-13 : IEC61131-3 変数と PFC 変数との相互関係

PFC 出力変数のアクセス用アドレス指定 (Index : インデックス、Sub-index : サブインデックス)

コントローラ内 PFC 出力変数の メモリ領域 (オフセット)	指定データ幅でアドレス指定															
	1 Byte		1 Word		3 Bytes		1 D. word		5 Bytes		6 Bytes		7 Bytes		8 Bytes	
	Index	Sub-index	Index	Sub-index	Index	Sub-index	Index	Sub-index	Index	Sub-index	Index	Sub-index	Index	Sub-index	Index	Sub-index
512	A000 A040 A080 *	LSB 1 MSB	A0C0 A100	LSB 1	A140 A180	LSB 1	A1C0 A200 A240	LSB 1	A280 A2C0	LSB 1	A300 A340	LSB 1	A380 A3C0	LSB 1	A400 A440	LSB 1
513	A000 A040 A080 *	2		MSB												
514	A000 A040 A080 *	3	A0C0 A100	2	A140 A180	MSB	A1C0 A200 A240	MSB	A280 A2C0	MSB	A300 A340	MSB	A380 A3C0	MSB	A400 A440	MSB
515	etc.															
516	.	.	etc.		A140 A180	2	A1C0 A200 A240	2	A280 A2C0	2	A300 A340	2	A380 A3C0	2	A400 A440	2
517	.	.														
518	.	.	etc.		A140 A180	2	A1C0 A200 A240	2	A280 A2C0	2	A300 A340	2	A380 A3C0	2	A400 A440	2
519	.	.														
520	.	.	etc.		A140 A180	2	A1C0 A200 A240	2	A280 A2C0	2	A300 A340	2	A380 A3C0	2	A400 A440	2
521	.	.														
522	.	.	etc.		A140 A180	2	A1C0 A200 A240	2	A280 A2C0	2	A300 A340	2	A380 A3C0	2	A400 A440	2
523	.	.														
524	.	.	etc.		A140 A180	2	A1C0 A200 A240	2	A280 A2C0	2	A300 A340	2	A380 A3C0	2	A400 A440	2
525	.	.														
526	.	.	etc.		A140 A180	2	A1C0 A200 A240	2	A280 A2C0	2	A300 A340	2	A380 A3C0	2	A400 A440	2
527	.	.														
528	.	.	etc.		A140 A180	2	A1C0 A200 A240	2	A280 A2C0	2	A300 A340	2	A380 A3C0	2	A400 A440	2
...	...	...														
1022			max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index
1023	max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index	max. Index	max Sub- Index

*最大 8Boolean 値が 1 バイト (1 サブインデックス) に格納できます。サブインデックスはビットに分解できません。

PFC 入力変数のアクセス用アドレス指定 (Index : インデックス、Sub-index : サブインデックス)

コントローラ内 PFC 入力変数の メモリ領域 (オフセット)	指定データ幅でアドレス指定															
	1 Byte		1 Word		3 Bytes		1 D. word		5 Bytes		6 Bytes		7 Bytes		8 Bytes	
	Index	Sub-index	Index	Sub-index	Index	Sub-index	Index	Sub-index	Index	Sub-index	Index	Sub-index	Index	Sub-index	Index	Sub-index
512	A480 A4C0 A500 *	LSB 1 MSB	A540 A580	LSB 1	A5C0 A600	LSB 1	A640 A680 A6C0	LSB 1	A700 A740	LSB 1	A780 A7C0	LSB 1	A800 A840	LSB 1	A880 A8C0	LSB 1
513	A480 A4C0 A500 *	2		MSB		MSB		MSB		MSB		MSB		MSB		MSB
514	A480 A4C0 A500 *	3	A540 A580	2	A5C0 A600	2	A640 A680 A6C0	2	A700 A740	2	A780 A7C0	2	A800 A840	2	A880 A8C0	2
515	etc.															
516	.		etc.		A5C0 A600	2	A640 A680 A6C0	2	A700 A740	2	A780 A7C0	2	A800 A840	2	A880 A8C0	2
517	.															
518	.		etc.		etc.		etc.		A700 A740	2	A780 A7C0	2	A800 A840	2	A880 A8C0	2
519	.															
520	.		etc.		etc.		etc.		A700 A740	2	A780 A7C0	2	A800 A840	2	A880 A8C0	2
521	.															
522	.		etc.		etc.		etc.		A700 A740	2	A780 A7C0	2	A800 A840	2	A880 A8C0	2
523	.															
524	.		etc.		etc.		etc.		A700 A740	2	A780 A7C0	2	A800 A840	2	A880 A8C0	2
525	.															
526	.		etc.		etc.		etc.		A700 A740	2	A780 A7C0	2	A800 A840	2	A880 A8C0	2
527	.															
528	.		etc.		etc.		etc.		A700 A740	2	A780 A7C0	2	A800 A840	2	A880 A8C0	2
...	...	...														
1022			max. Index	max Sub- In- dex	max. Index	max Sub- In- dex	max. Index	max Sub- In- dex	max. Index	max Sub- In- dex	max. Index	max Sub- In- dex	max. Index	max Sub- In- dex	max. Index	max- Sub- Index
1023	max. Index	max Sub- In- dex														

*最大 8Boolean 値が 1 バイト (1 サブインデックス) に格納できます。サブインデックスはビットに分解できません。

PFC フィールドバス変数の定義方法の例

以下の例は、オブジェクトディクショナリの特典入力項目に対比して、様々なデータタイプを持った PFC 変数に対する定義を示しています。

PFC 変数の データタイ プ	PFC 入力変数		PFC 出力変数	
	IEC61131-3 による定義	インデックス/ サブインデッ クス	IEC61131-3 による定義	インデックス/ サブインデッ クス
Unsigned8	InByte0 AT%IB512:BYTE; InByte0 AT%IB513:BYTE;	0xA4C0/1 0xA4C0/2	OutByte0 AT%QB512:BYTE; OutByte0 AT%QB513:BYTE;	0xA040/1 0xA040/2
Integer16	InInt0 AT%IW256:INT; InInt0 AT%IW257:INT;	0xA540/1 0xA540/2	OutInt0 AT%QW256:INT; OutInt0 AT%QW257:INT;	0xA0C0/1 0xA0C0/2
Unsigned16	InWord0 AT%IW256:WORD; InWord0 AT%IW257:WORD;	0xA580/1 0xA580/2	OutWord0 AT%QW256:WORD; OutWord0 AT%QW257:WORD;	0xA100/1 0xA100/2
Unsigned32	InDWord0 AT%ID128:DWORD; InDWord0 AT%ID129:DWORD;	0xA680/1 0xA680/2	OutDWord0 AT%QD128:DWORD; OutDWord0 AT%QD129:DWORD;	0xA200/1 0xA200/2

表 3-5 : オブジェクトディクショナリの特典入力項目に対する変数定義の例

3.1.6.4.1.4 最大インデックス

インデックスおよびサブインデックスの最大値は、512 バイトのフィールドバスコントローラのメモリサイズおよび個々のデータタイプのデータ幅によって決まります。

以下の表は、IEC61131-3 変数に対応する最大インデックスと最大サブインデックスの各値を示しています。

データタイプ	IEC61131-3 入力変数		IEC61131-3 出力変数	
	最大 インデックス	最大 サブインデックス	最大 インデックス	最大 サブインデックス
Integer8	0xA002	2	0xA482	2
Unsigned8	0xA042	2	0xA4C2	2
Boolean	0xA082	2	0xA502	2
Integer16	0xA0C1	1	0xA541	1
Unsigned16	0xA101	1	0xA581	1
Integer24	0xA140	AA	0xA5C0	AA
Unsigned24	0xA180	AA	0xA600	AA
Integer32	0xA1C0	80	0xA640	80
Unsigned32	0xA200	80	0xA680	80
Float32	0xA240	80	0xA6C0	80
Unsigned40	0xA280	66	0xA700	66
Integer40	0xA2C0	66	0xA740	66
Unsigned48	0xA300	55	0xA780	55
Integer48	0xA340	55	0xA7C0	55
Unsigned56	0xA380	49	0xA800	49
Integer56	0xA3C0	49	0xA840	49
Integer64	0xA400	40	0xA880	40
Unsigned64	0xA440	40	0xA8C0	40

表 3-6 : IEC61131-3 変数定義に対する最大インデックスおよび最大サブインデックス

例：

512 バイトの出力変数は、ワード単位 **Unsigned16** データタイプでアドレス指定されます。
この場合以下のように、256 ワード毎のデータアクセスが行われます。

- ーインデックス 0xA580、サブインデックス 1～255
- ーインデックス 0xA581、サブインデックス 1

インデックス	サブインデックス	内容	説明
0xA580	0	255	出力変数ブロック数
	1	D1*	1 番目変数ブロック
	...	...	...
	255	D255*	255 番目変数ブロック
	0	1	出力変数ブロック数
	1	D256*	256 番目変数ブロック

*D1=ワードデータ出力変数 1、D255=ワードデータ出力変数 255、他

3.1.6.4.2 絶対アドレス

CPU は絶対アドレスを用いて、モジュールデータに直接アクセスすることができます。アドレス指定は入力・出力両方共アドレス 0 から始まります。ビット、バイト、ダブルワード (DWORD) に相当するアドレスは、ワードアドレスから取り出されます。プロセスイメージの構造は 3.1.5 節「プロセスイメージ」に記載されています。アドレス指定は以下のような構造でおこなわれます。

入力データ	%IW0	ワード型データ
	%IW _n	
	%I _{n+1}	ビット型データ
	%I _{n+m}	
出力データ	%QW0	ワード型データ
	%QW _n	
	%Q _{n+1}	ビット型データ
	%Q _{n+m}	

3.1.6.4.3 アドレスの計算

ワードアドレスが計算上の基礎になります (WORD)。

ビットアドレス	ワードアドレス.0～.15
バイトアドレス	1 バイト目： 2×ワードアドレス
	2 バイト目： 2×ワードアドレス+1
ダブルワードアドレス	下位：ワードアドレス (偶数) /2
	上位：ワードアドレス (奇数) /2、少数以下切捨て

3.1.6.4.3.1 I/O モジュールデータのアドレス範囲

データ幅	I/O モジュールデータ用アドレス範囲								
ビット	0.0	0.8	1.0	1.8		254.0	254.8	255.0	255.8
	… 0.7	… 0.15	… 1.7	… 1.15		… 254.7	… 254.15	… 255.7	… 255.15
バイト	0	1	2	3		508	509	510	511
ワード	0		1		…	254		255	
ダブルワード	0					127			

3.1.6.4.3.2 フィールドバス変数のアドレス範囲

データ幅	フィールドバス変数用アドレス範囲								
ビット	256.0	256.8	257.0	257.8		510.0	510.8	511.0	511.8
	… 256.7	… 256.15	… 257.7	… 257.15		… 510.7	… 510.15	… 511.7	… 511.15
バイト	512	513	514	515		1020	1021	1022	1023
ワード	256		257		…	510		511	
ダブルワード	0					127			

3.1.6.4.3.3 フラグのアドレス範囲

データ幅	フラグ用アドレス範囲								
ビット	0.0	0.8	1.0	1.8		4094.0	4094.8	4095.0	4095.8
	… 0.7	… 0.15	… 1.7	… 1.15		… 4094.7	… 4094.15	… 4095.7	… 4095.15
バイト	0	1	2	3		8188	8189	8190	8191
ワード	0		1		…	4094		4095	
ダブルワード	0					2047			

全てのフラグは不揮発性です（リテイン）。

3.1.6.4.3.4 絶対アドレスの例

データ幅	入 力			
ビット	%IX14.0...15		%IX15.0...15	
バイト	%IB28	%IB29	%IB30	%IB31
ワード	%IW14		%IW15	
ダブルワード	%ID7			

データ幅	出 力			
ビット	%QX5.0...15		%QX6.0...15	
バイト	%QB10		%QB11	%QB12
ワード	%QW5		%QW6	
ダブルワード	%QD2（上位）		%QD3（下位）	

データ幅	フラグ			
ビット	%MX11.0...15		%MX12.0...15	
バイト	%MB22		%MB23	%MB24
ワード	%MW11		%MW12	
ダブルワード	%MD5（上位）		%MD6（下位）	

ビット指定用の文字'X'は省略できます（例：%I14.0、%Q6.10、%M11.7）。

3.1.7 WAGO-I/O-PRO CAA による PFC のプログラミング

CANopen フィールドバスコントローラは IEC61131-3 でプログラミングできるため、ユーザはフィールドバスコントローラの機能の他に PLC 機能を持つことを選択できます。IEC61131-3 に基づいたアプリケーションプログラムはプログラミングツールの WAGO-I/O-PRO CAA を用いて作成します。

本マニュアルには、WAGO-I/O-PRO CAA でプログラムする方法についての記述は入っておりません。その替わり以降の章では、ユーザが CANopen フィールドバスコントローラをプログラムする際確実に利用できるように、WAGO-I/O-PRO CAA のための特殊モジュールを説明することにします。

また本文では、IEC61131-3 プログラムをコントローラに転送することと、適切な通信ドライバをロードすることを説明します。



詳細情報

ソフトウェアの使用方法についての詳細は、WAGO-I/O-PRO CAA マニュアルを参照してください（ダウンロード：http://www.wago.co.jp/io/download_sitemap.html）。

3.1.7.1 CANopen 用 WAGO-I/O-PRO CAA ライブラリ要素

WAGO-I/O-PRO CAA には、色々な IEC61131-3 プログラミングアプリケーション用に様々なライブラリが提供されています。その中には汎用的な使用のためのモジュールがあり、それによってプログラムの作成が容易になり、またスピードアップすることができます。標準として 'standard.lib' ライブラリが用意されています。

次に示したライブラリは、WAGO-I/O-PRO CAA を特に CANopen プロジェクト用に使う場合を意図しています。

- “CIA405.LIB”
(CANopen や CiA405 準拠の CANopen 用のファンクションブロックが含まれます)

このライブラリは WAGO-I/O-PRO CAA の CD に入っています。
このライブラリを組み込むと、対象 POU、データタイプ、グローバル変数にアクセスし、自分で定義したのと同じように使用することができます。



詳細情報

POU およびソフトウェア操作についての詳細は、WAGO-I/O-PRO CAA マニュアルを参照してください（ダウンロード：http://www.wago.co.jp/io/download_sitemap.html）。

3.1.7.1.1 CIA405.LIB

要 素	説 明
CIA405_VERSION	POU は現ライブラリバージョン番号を返します
CIA405_GET_KERNEL_STATUS	機能モジュールは CANopen ノードのステータスを返します
CIA405_GET_LOCAL_NODE_ID	POU は自己モジュール ID を返します
CIA405_GET_STATE	POU は選択モジュールまたは自己モジュールの CANopen ネットワーク状態を返します
CIA405_RECV_EMY	POU はエマージェンシメッセージを受信したかどうかをチェックします。受信は自己ノードのモジュールから、または予め NMT_ADDNODE で構成したモジュールからのみ可能です。EXCEPTION_HANDLER は、受信したエマージェンシメッセージの数がシステムにバッファできるメッセージ数を越えたときを示します。
CIA405_RECV_EMY_DEV	POU はエマージェンシメッセージを特定のモジュールから受信したかどうかをチェックします。受信は自己ノードのモジュールから、または予め NMT_ADDNODE で構成したモジュールからのみ可能です。EXCEPTION_HANDLER は、受信したエマージェンシメッセージの数がシステムにバッファできるメッセージ数を越えたときを示します。
CIA405_SDO_READ4	POU はオブジェクトディクショナリの入力値を返します (最大データ長=4 バイト)。
CIA405_SDO_READ7	POU はオブジェクトディクショナリの入力値を返します (最大データ長=7 バイト)。
CIA405_SDO_READ21	POU はオブジェクトディクショナリの入力値を返します (最大データ長=21 バイト)。
CIA405_SDO_READ210	POU はオブジェクトディクショナリの入力値を返します (最大データ長=210 バイト)。
CIA405_SDO_WRITE4	POU は規定したデータをあるモジュールのオブジェクトディクショナリに書き込みます (最大データ長=4 バイト)。
CIA405_SDO_WRITE7	POU は規定したデータをあるモジュールのオブジェクトディクショナリに書き込みます (最大データ長=7 バイト)。
CIA405_SDO_WRITE21	POU は規定したデータをあるモジュールのオブジェクトディクショナリに書き込みます (最大データ長=21 バイト)。
CIA405_SDO_WRITE210	POU は規定したデータをあるモジュールのオブジェクトディクショナリに書き込みます (最大データ長=210 バイト)。

要 素	説 明
EXCEPTION_HANDLER	POU は起きたエラーを返し、直接エラー修復をすることを許可します。
NMT_ADDNODE	POU は規定したモジュールに対して、ガーディングタイムとライフタイムファクタに関するガーディングオブジェクトをイニシャライズします。
NMT_CHANGE_STATE	POU は予め NMT_ADDNODE でイニシャライズされたモジュールのステータス変更をトリガします
NMT_DELNODE	POU は取り扱われるスレーブのリストから規定したノードを削除します。
NMT_GUARDING	POU はガーディング監視の ON、OFF を切り替えます。
NMT_SLAVE_IN_NE_LIST	POU は規定したノードが内部スレーブリストにあるかどうかのメッセージを返します。
SEND_CAN_FRAME	POU は CAN バス経由で、規定した COB_ID の与えられたデータを送信します（最大データ長=8 バイト）。
SYNC_MASTER_SEND_SYNC	POU は SYNC メッセージを送信します。

3.1.7.2 IEC61131-3 プログラム転送

目的の IEC61131-3 アプリケーションのプログラミングに続いて、PC からコントローラへのプログラム転送は 2 種類の異なった方法で行うことができます。

- シリアルインタフェース経由
- フィールドバス経由

上記いずれの方法に対しても、各々適切な通信ドライバが必要となります。



詳細情報

通信ドライバのインストール方法およびソフトウェアの使用法についての詳細は、WAGO-I/O-PRO CAA マニュアルを参照してください（ダウンロード：http://www.wago.co.jp/io/download_sitemap.html）。

3.1.7.2.1 シリアルインタフェースによる転送

シリアルインタフェースとの物理的接続にはワゴ通信ケーブルを使用します。これは IEC 61131-3 用プログラミングツール (型番: 759-333) に付属していますが、アクセサリとして個別に購入することもできます (型番: 750-920)。

パソコンの COMX ポートとコントローラの通信インタフェースをワゴ通信ケーブルで接続します。



注意

通信ケーブル 750-920 は、コントローラの電源が入っている間は抜き差ししないでください。

シリアルデータ転送には通信用ドライバが必要です。**WAGO-I/O-PRO CAA** では、このドライバとそのパラメータは[**Communication parameters**]ダイアログで入力します。

1. Windows の[**スタート**]メニューから **CoDeSys V2.3** (WAGO Software/CoDeSys for Automation Alliance の下) を選ぶか、またはデスクトップ上の **CoDeSys V2.3** アイコン (またはショートカット) をダブルクリックして起動します。
2. [**Online**]メニューにおいて [**Communication parameters**]をクリックすると、[**Communication parameters**]のダイアログウインドウが開きます。新しい通信チャンネルを作るために[**New**]をクリックすると、[**Communication Parameters: New Channel**]のダイアログウインドウが開きます。
3. [**Communication Parameters: New Channel**]のダイアログウインドウでは[**Name**]フィールドでチャンネル名が指定できます。次に[**Device**]欄で[**Serial(RS232)**]をクリックします。続けて[**OK**]をクリックすると、このダイアログウインドウが閉じ、[**Communication parameters**]ダイアログウインドウが再び前面に表示されます。
4. ダイアログの中央ウインドウには次に示すパラメータが表示されます。
Port=COM1
Baudrate=19200
Parity=Even
Stop bits=1
Motorola byteorder=No
必要があれば値を変更します。変更がすべて終わったら[**OK**]をクリックします。これでコントローラと通信が確立し、テストをスタートできます。



メモ

コントローラにアクセスするには、コントローラの動作モードスイッチが中央位置または最上位置にある必要があります。

5. [**Online**]メニューにおいて[**Login**]をクリックし、コントローラにログインします (オンライン操作中は **WAGO-I/O-PRO CAA** サーバはアクティブ状態にあり、[**Communication parameters**]のメニュー項目は使用できません)。
6. コントローラにアプリケーションプログラムがないとき、または異なるプログラムが入っているときは、ダイアログウインドウが開いて新しいプログラムをロードするかどうか聞いてきます。ここで[**Yes**]を選択すると、現在のプログラムがロードされます。
7. プログラムのロードが完了したら、[**Online**]メニューの[**Run**]をクリックすることによってプログラムをスタートすることができます。このときステータスバー (画面の最下部にあります) の右端にある **RUNNING** の表示が強調表示に変わります。
8. オンライン操作を終了するには、[**Online**]メニューの[**Logout**]をクリックします。

3.1.7.2.2 フィールドバスによる転送

1. Windows の[スタート]メニューから **CoDeSys V2.3** (WAGO Software/CoDeSys for Automation Alliance の下) を選ぶか、またはデスクトップ上の **CoDeSys V2.3** アイコン (またはショートカット) をダブルクリックして起動します。
2. **[Online]**メニューにおいて **[Communication parameters]** をクリックすると、**[Communication parameters]** のダイアログウィンドウが開きます。
3. 新しい通信チャンネルを作るために **[New]** をクリックすると、**[Communication Parameters: New Channel]** のダイアログウィンドウが開きます。
4. このダイアログウィンドウ内の **[Name]** フィールドでは任意のチャンネル名を入力し、次に **[Device]** 欄で必要なドライバ (例: **CANopenHilscher**) を選択します。
5. 続けて **[OK]** をクリックすると、このダイアログウィンドウが閉じ、以下の項目がダイアログの中央ウィンドウに表示されます。
 - Board number (使用 PC 構成部品に応じた **CANopen** フィールドバス PCB 番号)
 - Baudrate (コントローラの DIP スイッチで設定したボーレート)
 - Node ID (コントローラの DIP スイッチで設定したモジュール ID)
 - Database、Debug level など必要に応じて上記の設定値を変更します。
6. データベースから以前の設定を消去したい場合は、項目 **Database** の **Value** 値を **RETAIN** から **CLEAR** に変更することができます。
7. **[OK]** を押すとダイアログウィンドウが閉じます。
8. **[Online]**メニューにおいて再度 **[Communication parameters]** をクリックすると、**[Communication parameters]** のダイアログウィンドウが開きますので、項目 **Database** の **Value** 値を **CLEAR** から **RETAIN** に戻します。
9. ここで再び **[OK]** を押すとダイアログウィンドウが閉じますので、コントローラのテストを開始することができます。



メモ

コントローラにアクセスするための必要条件是、モジュール ID とボーレートがコントローラの設定と一致していること、およびコントローラの動作モードスイッチが中央位置または最上位置にあることです。

10. **[Online]**メニューにおいて **[Login]** をクリックし、コントローラにログインします (オンライン操作中は **WAGO-I/O-PRO CAA** サーバはアクティブ状態にあり、**[Communication parameters]** のメニュー項目は使用できません)。
11. コントローラにアプリケーションプログラムがないとき、または異なるプログラムが入っているときは、ダイアログウィンドウが開いて新しいプログラムをロードするかどうか聞いてきます。ここで **[Yes]** を選択すると、現在のプログラムがロードされます。
12. プログラムのロードが完了したら、**[Online]**メニューの **[Run]** をクリックすることによってプログラムをスタートすることができます。このときステータスバー (画面の最下部にあります) の右端にある **RUNNING** の表示が強調表示に変わります。
13. オンライン操作を終了するには、**[Online]**メニューの **[Logout]** をクリックします。

3.1.8 CANopen フィールドバスノードの立ち上げ

この章では、ワゴ CANopen フィールドバスノードを立ち上げるときの手順を、ステップを追って説明します。



注意

この説明は一例として示され、個別の CANopen フィールドバスノードをローカルで立ち上げる場合に限られます。

手順は以下のステップからなります。

1. PC とフィールドバスノードの接続
2. 現在ボーレートの確認と新規のボーレートの設定
3. モジュール ID の設定
4. OPERATIONAL ステータスに移行
5. アナログ入力データのオン設定
6. アプリケーション固有のマッピング

3.1.8.1 PC とフィールドバスノードの接続

CANopen フィールドバスノードをフィールドバスケーブルによって PC 内の CANopen フィールドバス PCB に接続し、PC を立ち上げます。

3.1.8.2 現在ボーレートの確認と新規のボーレートの設定

最初に全ての DIP スイッチを”OFF”位置（モジュール ID=0）にして、フィールドバスコントローラに電源（DC24V 電源）を投入します。



図 3-14 : ボーレートの確認と設定=全ての DIP スイッチを”OFF”

g012442x

このとき現在のボーレートは、上側 4 個の LED が点滅表示することにより確認することができます。



注意

DIP スイッチの一部が”OFF”位置にない場合に電源を投入すると、現在の設定内容がモジュール ID として書き込まれることになります。

ここで必要なボーレートに対応する DIP スイッチを”ON”にしてください。

例：DIP スイッチ 3 を”ON”にするとボーレート 125kB になります。



図 3-15 : ボーレート 125kB の設定

g012443x

新しい設定内容を保存するためには DIP スイッチ 8 を”ON”にしてください。

その後コントローラの電源を切ります。



図 3-16 : ボーレート 125kB の保存

g012441x

3.1.8.3 モジュール ID の設定

モジュール ID は電源を切断して設定します。設定するときは、全ての DIP スイッチを一旦”OFF”位置にしてください。それから必要なモジュール ID に相当する DIP スイッチを”ON”にしてください。例えば、モジュール ID 1 を設定するには DIP スイッチ 1 を”ON”にします。

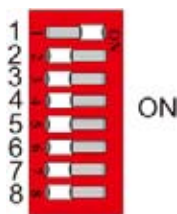


図 3-17 : モジュール ID 1 の設定

g012440x

電源を投入後すぐに、コントローラは INITIALIZATION ステータスに入ります。同時に、プロセスイメージは、アプリケーション固有のコンフィグレーションが保存されていない場合、接続されたバスモジュールとデフォルトマッピング後にイニシャライズされたオブジェクトディクショナリを用いて生成されます。

イニシャライゼーション・フェーズがエラー無しで終了した後、コントローラは自動的に PRE-OPERATIONAL ステータスに移行します。

このステータスでは通信は SDO を介して行うことができます。このときユーザは、CAN マスタソフトウェア経由で SDO を続けて用いることにより、さまざまな設定を行うことができます。

- コントローラは直接 OPERATIONAL ステータスにセットすることができます。



メモ

デフォルト設定では「アナログ入力データの PDO 送信を遮断します」と定義されていますので、アナログ入力データは読むことができません。

- ユーザはアナログ入力データの送信遮断状態を解放するか、またはアプリケーション固有のマッピングを選択できます。

3.1.8.4 OPERATIONAL ステータスへの移行

ネットワークマネジメント・オブジェクトから”リモートノードスタート”コマンドを用いて、コントローラを PRE-OPRRATIONAL ステータスから OPERATIONAL ステータスに移行することができます。これによりフィールドバスノードは PDO の通信が可能になります（4.3.3.1.1 節「リモートノードスタート」を参照してください）。



メモ

デフォルト設定として、アナログ入力データの PDO 送信はオフになっています。これにより、このデータは 1 回読み出されるだけで、その後変更されません。PDO 経由でこのデータを利用できるようにするために PRE-OPERATIONAL ステータスでアナログ入力データをオンにします。SDO 経由のアクセスはいつでも可能です。

他に設定操作をしなければコントローラは動作状態となります。通信はデフォルトマッピングに従って可能となります（4.3.4.1 節「イニシャライゼーション」を参照してください）。

3.1.8.5 アナログ入力データのオン設定

CAN バスが CAN メッセージでオーバーフローするのを避けるために、PDO によるアナログ入力データの送信は、デフォルト設定で停止状態（deactivate）になっています。これはオブジェクト 0x6423 ”アナログ入力グローバルインタラプト・イネーブル”のデフォルト値が”FALSE”（=0）であることを意味します（4.3.4.4.12 節「オブジェクト 0x6423 ”アナログ入力グローバルインタラプト・イネーブル”」を参照してください）。

コントローラが PRE-OPERATIONAL ステータスにいるときオブジェクト 0x6423 を TRUE（=1）値にセットすることにより、一般的に送信をすることができます。この後、”リモートノードスタート”コマンドを用いることにより、コントローラのステータスを PRE-OPRRATIONAL から OPERATIONAL に変更することができます。この処理により、PDO 経由の通信とアナログ入力データの送信が可能となります。

他に設定処理をしなければコントローラは動作可能となり、通信はデフォルトマッピングに従って可能となります（4.3.4.1 節「イニシャライゼーション」を参照してください）。

3.1.8.6 アプリケーション固有のマッピング

デフォルトマッピングを変更するときは、アプリケーション固有の PDO のマッピングに PDO で送信するデータを定義します。このためにコントローラは PRE-OPERATIONAL ステータスにしなければなりません。

アプリケーション固有のマッピングを行う方法の詳細を以下に説明します。

例：

3 番目及び 5 番目の 2 バイトアナログ入力チャンネルと 1 番目の 8 ビットデジタル入力のグループは TxPDO 2 を用いて読むことにします。ここでは送信用に CAN ID（識別子）0x432 を用います。送信は 3 番目の SYNC オブジェクト毎に同期して行わなければなりません。デフォルトの CAN ID は SDO 用に用いられます。ノード 8 で設定します。xx...は無視されます。

1. 最初に、インデックス 0x1A01、サブインデックス 0 のマッピングオブジェクトの数を 0 にすることにより PDO マッピングを停止状態にします（送信 PDO マッピングパラメータ）。

PDO マッピングを動作停止にする		
	CAN ID	データ
送信	608	0x2F 01 1A 00 00 xx xx xx
受信	588	0x60 01 1A 00 xx xx xx xx

2. TxPDO マッピングパラメータ構造 (インデックス 0x1A01) にインデックス、サブインデックス、アプリケーションオブジェクトのオブジェクト長を入れます。1 個の PDO 当たり最大 8 バイトのデータをアサインすることができます。

マッピングパラメータ構造に以下を書き込む		
アプリケーションオブジェクト	インデックス	サブインデックス
3. アナログ入力チャンネル	0x6401	3
5. アナログ入力チャンネル	0x6401	5
1. デジタル入力グループ	0x6000	1

3. 一連のタスクを保証する為に 2 番目の TxPDO マッピングパラメータは以下の構造にならなければなりません。

TxPDO マッピングパラメータ構造、インデックス 0x1A01			
サブインデックス	アプリケーションオブジェクト		オブジェクト長 (ビット単位)
	インデックス	サブインデックス	
0	3		
1	0x6401	3	0x10
2	0x6401	5	0x10
3	0x6000	1	0x08



メモ

最初にサブインデックス 0 にマッピングパラメータのサブインデックス 1~8 を入れ、次に有効なサブインデックスの数を入れてください。

これらのオブジェクトは SDO の送信によって格納されます。

マッピング 3. アナログ入力チャンネル		
	CAN ID	データ
送信	0x608	0x23 01 1A 01 10 03 01 64 23 データが 0 の場合無効 01 1A インデックス (下位バイトが先) 01 サブインデックス 10 アナログチャンネルのデータ長 03 サブインデックス、3 番目のアナログチャンネルは製造業者プロファイルにある 01 64 インデックス (下位バイトが先)、3 番目のアナログチャンネルは製造業者プロファイルにある
受信	0x588	0x60 01 1A 01 xx xx xx xx 60 OK 01 1A インデックス (下位バイトが先) 01 サブインデックス

マッピング 5. アナログ入力チャンネル		
	CAN ID	データ
送信	0x608	0x23 01 1A 02 10 05 00 64
受信	0x588	0x60 01 1A 02 xx xx xx xx

マッピング 1. デジタル入力グループ		
	CAN ID	データ
送信	0x608	0x23 01 1A 03 08 01 00 60
受信	0x588	0x60 01 1A 03 xx xx xx xx

マッピングオブジェクト数=3、サブインデックス 0 に入れる		
	CAN ID	データ
送信	0x608	0x2F 01 1A 00 03 xx xx xx
受信	0x588	0x60 01 1A 00 xx xx xx xx

4. ここで通信パラメータを変更するためにマッピングする PDO を動作停止にします。現在の例では TxPDO2 になります。このためにはインデックス 0x1801、サブインデックス 01 のオブジェクトに 0x80000000 の値を書き込みます（送信 PDO 通信パラメータ）。

PDO マッピングを動作停止にする		
	CAN ID	データ
送信	608	0x23 01 18 01 00 00 00 80
受信	588	0x60 01 18 01 xx xx xx xx

5. インデックス 0x1801 オブジェクト（送信 PDO 通信パラメータ）のサブインデックス 1～3 に構造体の通信パラメータを書き込みます。ここでは送信タイプは 3（各々 3 番目の SYNC オブジェクト毎に同期して送信する）。

通信パラメータの入力		
TxPDO 通信パラメータ、インデックス 0x1801		
サブインデックス	値	意味
0	3	レコード内でサポートする入力項目数
1	0x432	PDO で使用する COB-ID
2	3	送信タイプ
3	0	Inhibit 時間

サブインデックス 3: Inhibit 時間=0		
	CAN ID	データ
送信	0x608	0x2B 01 18 03 00 00 xx xx
受信	0x588	0x60 01 18 03 xx xx xx xx

サブインデックス 2: 送信タイプ=3		
	CAN ID	データ
送信	0x608	0x2F 01 18 02 03 xx xx xx
受信	0x588	0x60 01 18 02 xx xx xx xx

サブインデックス 1: PDO を COB-ID=432 に変更し、PDO を無効→有効にする		
	CAN ID	データ
送信	0x608	0x2B 01 18 01 32 04 xx xx
受信	0x588	0x60 01 18 01 xx xx xx xx

6. "リモートノードスタート"メッセージを用いてバスコントローラを OPERATIONAL に変更すると PDO は活動状態（activated）になり、TxPDO オブジェクトはデータ送信に使用できるようになります。

3.1.9 LED 表示

コントローラ上の複数の LED は、コントローラの動作状態またはノードの正常性を表示します。

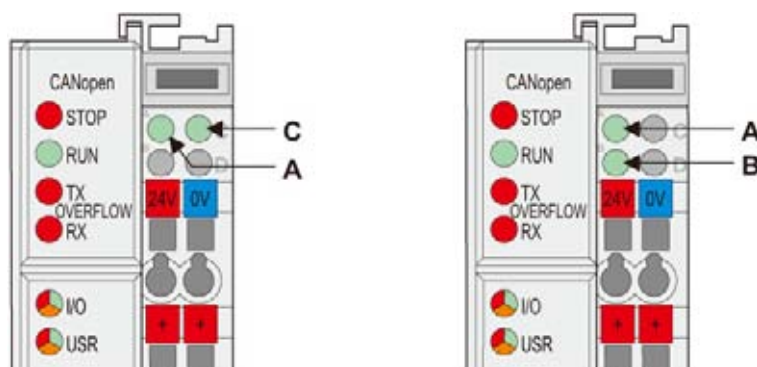


図 3-18 : 表示素子

LED は 3 グループに分かれます。

グループ 1=フィールドバスの状態を示します。STOP (赤)、RUN (緑)、Tx-Overflow (赤)、Rx-Overflow (赤) があり、CAN ラインの通信動作状態を知らせます。

グループ 2=内部バス状態を示します。I/O-LED は 3 色 (赤/緑/橙) で、内部バス状態やソフトウェア上のエラーコードを知らせます。

最下部の USR-LED は 3 色で、プログラマブルバスコントローラ内のユーザプログラムでアクセスすることができます。

グループ 3=電源の状態を示します。コントローラ上部右上の A、B または A、C の緑色 LED があり、システム電源およびフィールド電源の状態を知らせます。

3.1.9.1 フィールドバスステータス

上部 4 個の LED (STOP、RUN、Tx-Overflow、Rx-Overflow) は CAN 通信の動作状態を表します。

STOP	RUN	Tx-Overflow	Rx-Overflow	意味	対処方法
オフ	オフ	オフ	オフ	動作不能または自己テスト中	電源をチェック (24V 及び 0V)、自己テストを待つ
オフ	遅い点滅	X	X	モジュールは PRE-OPERATIONAL 状態である	
オフ	オン	X	X	モジュールは OPERATIONAL 状態である	
オン	オフ	X	X	モジュールは STOP 状態である、又はフィールドバスに無関係の重大なエラー (モジュールが除去されたなど)、不正なコンフィグレーション	フィールドバスに無関係のエラー内容を調べる、ノードをリセットする、コンフィグレーションエラーの内容を調べる
X	X	X	オン	CAN 受信バッファがフル、データ欠損の可能性あり	2 つのプロトコルの時間間隔を伸ばす
X	X	オン	X	CAN 送信バッファがフル、データ欠損の可能性あり	バスシステムのデータ長をチェックする、モジュールの送信優先度を上げる
X	X	高速点滅、Rx-Overflow 点滅を伴う	高速点滅、Tx-Overflow 点滅を伴う	CAN コントローラが警告レベルを超えた、大量のエラーメッセージ	ボーレートとバス接続を確認、ネットワークに最低 2 モジュール設置する
オフ	高速点滅	X	X	モジュールは PRE-OPERATIONAL 状態、Sync/Guard メッセージまたは Heartbeat の故障	OPERATIONAL 状態に変更し、Sync/Guard メッセージまたは Heartbeat を再スタートする
高速点滅	高速点滅	X	X	モジュールは OPERATIONAL 状態、Sync/Guard メッセージまたは Heartbeat の故障	Sync/Guard メッセージまたは Heartbeat を再スタートする
高速点滅	オフ	X	X	モジュールは STOP 状態、Sync/Guard メッセージまたは Heartbeat の故障	OPERATIONAL 状態に変更し、Sync/Guard メッセージまたは Heartbeat を再スタートする

3.1.9.2 ノードステータス—I/O LED の点滅コード

LED	色	意味
I/O	赤／緑／橙	「I/O」LED はノードの動作状態を示し、エラーの発生を知らせる

電源をオンにするとコントローラが始動し、「I/O」LED が赤く点滅します。コントローラが正常に立ち上がると、「I/O」LED は緑色に点灯します。異常が検出されると「I/O」LED が赤色に点滅し続けます。異常の内容は点滅パターンによって繰り返し表示されます。

詳細のエラーメッセージは I/O ERR LED の点滅で表示されます。エラー内容は、3 回までの点滅シーケンスにより周期的に表示されます。

- 最初的高速点滅シーケンス（約 10Hz）により、エラー表示のスタートが分かります。
- 2 番目の点滅シーケンス（約 1Hz）は休止の後に続いて行われます。点滅の回数により、エラーコードが表示されます。
- 3 番目の点滅シーケンス（約 1Hz）は再度の休止の後に行われます。点滅の回数により、エラー引数が表示されます。

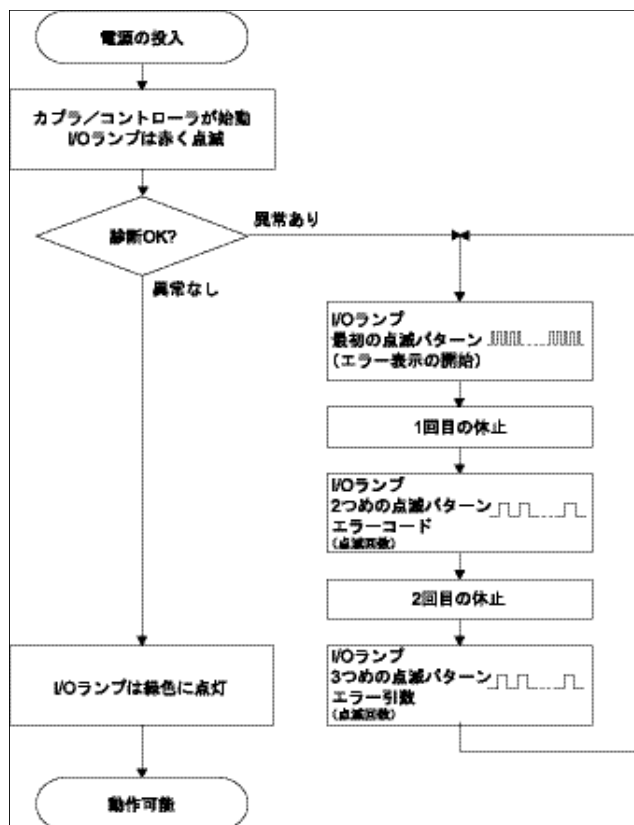


図 3-19 : I/O ERR LED によるノード状態の表示

g012111d

異常が解消されたら、電源を再投入してコントローラを再起動します。

I/O LED	意味
緑	内部バスにデータあり
消灯	内部バスにデータなし
赤	コントローラのハードウェア異常
赤 点滅	a) 起動時：内部バスの初期化 b) 動作中：一般的な内部バス異常
赤 周期的に点滅	内部バスリセットおよび内部異常時のエラーメッセージ エラーメッセージは 3 回の連続点滅シーケンスで表示されます。 各点滅シーケンスの間に短い休止時間があります。

I/O LED のエラーコード

- 1 回目の点滅シーケンス：エラーメッセージのスタート
- 2 回目の点滅シーケンス：エラーコード
- 3 回目の点滅シーケンス：エラー引数

エラーコード 1：ハードウェア障害およびコンフィグレーションエラー		
エラー引数	エラー内容	対処方法
-	パラメータデータのチェックサムエラー	ノードの電源を切り、I/O モジュール数を減らし、その後電源を再立ち上げします。
1	インラインコード用内部バッファメモリのオーバーフロー	ノードの電源を切り、I/O モジュール数を減らし、その後電源を再立ち上げします。エラーがなくならない場合はコントローラを交換します。
2	不明なデータタイプの I/O モジュールあり	エラーの I/O モジュールは以下のように検出してください。 ： 電源を切った後終端モジュールをノードの中間に置きます。電源を再投入します。 —LED がまだ点滅している場合電源を切り、終端モジュールをノードの前半分（コントローラに向かって）の中間に置きます。 —LED が点滅しない場合は電源を切り、終端モジュールをノードの後ろ半分（コントローラから離れて）の中間に置きます。 —電源を再投入し、この手順を故障 I/O が検出されるまで繰り返します。 —故障モジュールを交換します。 —コントローラに問題がある場合はファームウェアを更新するよう要請してください。
3	フラッシュプログラムメモリのモジュールタイプが判断できない、または誤り	ノードの電源を切り、コントローラを交換した後電源を再投入します。
4	フラッシュメモリに書き込み中のエラー	ノードの電源を切り、コントローラを交換した後電源を再投入します。
5	フラッシュメモリの消去時のエラー	ノードの電源を切り、コントローラを交換した後電源を再投入します。
6	AUTORESET 後に I/O モジュールのコンフィグレーション変更が検出された	電源を一旦切り、再投入することによりコントローラを再立ち上げしてください。
7	EEPROM へのデータ書き込み時のエラー	ノードの電源を切り、コントローラを交換した後電源を再投入します。
8	ハードウェアとファームウェアのバージョンの不適合	ノードの電源を切り、コントローラを交換した後電源を再投入します。
9	EEPROM のチェックサムエラー	ノードの電源を切り、コントローラを交換した後電源を再投入します。
10	EEPROM の初期化エラー	ノードの電源を切り、コントローラを交換した後電源を再投入します。
11	EEPROM からのデータ読み出し時のエラー	ノードの電源を切り、コントローラを交換した後電源を再投入します。
12	EEPROM へのデータ書き込み時のタイムアウト	ノードの電源を切り、コントローラを交換した後電源を再投入します。
14	ゲートウェイまたはメールボックス I/O モジュールの接続最大数を越えた	ノードの電源を切り、ゲートウェイまたはメールボックス I/O モジュールの数を減らした後電源を再投入します。
エラーコード 2：未使用		

エラー引数	エラー内容	対処方法
—	未使用	—
エラーコード 3 : 内部バスのプロトコルのエラー		
エラー引数	エラー内容	対処方法
—	内部バス通信上の動作不良：故障モジュールは検出されない	ノード内に 750-613 電源入力モジュールが使われていた場合、このモジュールへの電源が正常に機能していることを最初に確認してください。これは LED の表示によって分かります。 全ての I/O モジュールが正しく接続されているかノードに 750-613 が使用されていない場合はエラー I/O モジュールを次のように検出することができます。 —ノードの電源を切り、ノードの中間に終端モジュールを置きます。電源を再投入します。 —LED がまだ点滅している場合電源を切り、終端モジュールをノードの前半分の中間に置きます。 —LED が点滅しない場合は電源を切り、終端モジュールをノードの後ろ半分の中間に置きます。 —電源を再投入し、この手順を故障 I/O が検出されるまで繰り返します。 —故障モジュールを交換します。 —モジュールが 1 個だけ残り、LED がまだ点滅している場合はこのモジュールまたはコントローラが故障しているとみなされます。 —当該モジュールを交換してください。
エラーコード 4 : 内部バスの物理的エラー		
エラー引数	エラー内容	対処方法
—	内部バスのデータ通信エラーまたはコントローラにおける内部バスの中断エラー	ノードの電源を切ります。プロセスデータを持った I/O モジュールをコントローラの後に置き、電源を入れた直後のエラー引数を読みます。 I/O LED でエラー引数が示されなかった場合はコントローラを交換してください。 これ以外の場合は次のようにエラー I/O を検出します。 —ノードの電源を切り、ノードの中間に終端モジュールを置きます。電源を再投入します。 —LED がまだ点滅している場合電源を切り、終端モジュールをノードの前半分の中間に置きます。 —LED が点滅しない場合は電源を切り、終端モジュールをノードの後ろ半分の中間に置きます。 —電源を再投入し、この手順を故障 I/O が検出されるまで繰り返します。 —故障モジュールを交換します。 —モジュールが 1 個だけ残り、LED がまだ点滅している場合はこのモジュールまたはコントローラが故障しているとみなされます。 —当該モジュールを交換してください。
n* (n>0)	n 番目の I/O モジュールの後段にて内部バスが中断した	ノードの電源を切り、n+1 番目の I/O モジュールを交換します。その後電源を再投入します。
エラーコード 5 : 内部バスの初期化時エラー		
エラー引数	エラー内容	対処方法
n*	内部バスの初期化中にレジスタ通信でエラーが起きた	ノードの電源を切り、n 番目の I/O モジュールを交換します。その後電源を再投入します。
エラーコード 6 : ノード設定時のコンフィグレーションエラー		
エラー引数	エラー内容	対処方法
n	スレーサモジュールのマッピング時 I/O モジュールのコンフィグレーション間違い	スレーサモジュールのマッピングを調べてください。
エラーコード 9 : CPU 例外エラー		
エラー引数	エラー内容	対処方法
1	不正なオペレーションコード	プログラムシーケンス・エラー。
2	スタック・オーバーフロー	ワゴジャパン、オートメーションに連絡してください。
3	スタック・アンダーフロー	
4	NMI	

エラーコード 10 : CPU プログラムエラー		
エラー引数	エラー内容	対処方法
1	デジタル入力のオフセットアドレス無効	適切なファンクションブロックでオフセットアドレスを修正します。
2	デジタル出力力のオフセットアドレス無効	
エラーコード 11 : メールボックス I/O モジュールエラー		
エラー引数	エラー内容	対処方法
1	メールボックス機能付きモジュールが最大接続数を超えた	ノードの電源を切り、メールボックス機能付きモジュール数を減らした後電源を再投入してください。
		メモ ファームウェアバージョンが 11 のコントローラに対しては、「メールボックス機能付きモジュール最大 3 個」が可能であり、ファームウェアバージョンが 12 以降のものに対しては、「メールボックス機能付きモジュール最大 8 個」が可能です。
		注意 アナログ I/O モジュールおよびメールボックス機能を持ったモジュールをコンフィグレーションするときは、プロセスイメージのサイズの区切りに注意してください。 ノードの全ての I/O モジュールを完全にコンフィグレーションすると、特に AS-Interface マスタ（データ幅最大 48 バイト）などは、場合によって既に内部限界に達してしまう可能性があります。このエラーはまた、プロセスイメージの最大値に到達していない場合にも起こりえます。
2	メールボックスが最大サイズを超えた	メールボックスのサイズを減らしてください。
3	メールボックス機能付きモジュールを置くことにより、プロセスイメージが最大サイズを超えた	メールボックス機能付きモジュールのデータ巾を減らしてください。

* 点滅回数 (n) は I/O モジュールの実装位置を表します。ただし、データの無い I/O モジュール（診断なしの電源入力モジュールなど）はカウントされません。

例 : 13 番目の I/O モジュールが取り外された。	
1.	「I/O」LED が最初の点滅パターン（毎秒約 10 回）によってエラー表示を開始。
2.	最初の休止に続いて、2 番目の点滅パターン（毎秒約 1 回）を表示。「I/O」LED は 4 回点滅し、エラーコードが 4（内部バスデータの異常）であることを表示。
3.	2 回目の休止に続いて、3 番目の点滅パターンを表示。「I/O」LED は 12 回点滅する。エラー引数「12」は、内部バスが 12 番目の I/O モジュールの後段で中断されたことを示す。

3.1.9.3 電源状態

LED	色	意味
A	緑	動作電圧の状態—システム
B または C	緑	動作電圧の状態—電源ジャンパ接点 (LED の位置は製造年度により異なります)

コントローラの電源部分には、供給電圧の状態を示す緑色の LED が 2 個あります。左の LED (A) はコントローラ用の 24V 給電を示し、下の LED (B) または右の LED (C) はフィールド側（電源ジャンパ接点）への給電を示します。

3.1.10 技術データ

システムデータ	
CAN ネットワークに接続可能な最大ノード数（リピータなし）	110
伝送媒体	シールド付き銅線ケーブル、 $3 \times 0.25 \text{ mm}^2$
バスラインの最大距離	30m～100m (ボーレートとケーブル種別に依存)
ボーレート	10kBaud～1MBaud
バスコントローラの接続 750-837、/020-000、/021-000 : 750-838、/020-000、/021-000 :	231 シリーズ (MCS) 5 極オスコネクタ、メスコネクタ 231-305/010-000 は標準装備 D-SUB9 ピン・オスコネクタ メスコネクタ 750-963 は別売
プログラミング	WAGO-I/O-PRO 32、CoDeSys V2.3
IEC 61131-3	IL、LD、FBD、ST、FC
規格および認定 (2.2 章参照)	
EMC CE イミューニティ規格 750-837、/020-000、/021-000 : 750-838、/020-000、/021-000 :	EN50082-2 (1996) に準拠 EN61000-6-2 (2001) に準拠
EMC CE エミッション規格 750-837、/020-000、/021-000 : 750-838、/020-000、/021-000 :	EN50081-2 (1994) に準拠 EN61000-6-3 (2001) に準拠
EMC 船舶使用 —イミューニティ規格	ドイツ船級協会規格 (01) に準拠 —750-837 のみ
EMC 船舶使用 —エミッション規格	ドイツ船級協会規格 (01) に準拠 —750-837 のみ
承認 (2.2 章参照)	
UL *)	E175119、UL508
DEMKO *)	02 ATEX 132273 X II 3 G EEx nA II T4
GL (ドイツ船級協会) *)	Cat. A, B, C, D
LR (英国ロイド船級協会) *)	02/20026 Env. 1, 2, 3, 4
DNV (ノルウェー船級協会) *)	A-9334 Cl. B
RINA (イタリア船級協会) *)	MAC30402CS1
ABS (米国船級協会) *)	03-HG374860-I/PDA
CE 準拠マーク *)	
*) 750-837 のみ	
アクセサリ	
EDS ファイル	750-914 : www.wago.co.jp/io/index.htm からダウンロード
ミニチュア WSB クイックマーキングシステム	

技術データ	
最大 I/O モジュール数	64
メールボックス機能付モジュール数	最大 3 (ファームウェアバージョン 11) 最大 8 (ファームウェアバージョン 12 以降)
フィールドバス ー入力プロセスイメージ ー出力プロセスイメージ ー入力変数 ー出力変数	最大 512 バイト 最大 512 バイト 最大 512 バイト 最大 512 バイト
データメモリ 750-837、-838 : 750-837/020-000、750-838/020-000 : 750-837/021-000、750-838/021-000 :	ー64k バイト ー192k バイト ー832k バイト
プログラムメモリ 750-837、-838 : 750-837/020-000、750-838/020-000 : 750-837/021-000、750-838/021-000 :	ー128k バイト ー256k バイト ー640k バイト
不揮発性メモリ	8k バイト (リテイン)
サイクルタイム	<3ms (1,000 ステートメント/256 デジタル I/O の場合)
使用可能 PDO 数	32Tx/32Rx
使用可能 SDO 数	2×サーバ SDO/16×クライアント SDO
通信プロファイル	DS-301 V4.01
デバイスプロファイル	DS-401 V2.0 : 限界チェック、エッジトリガ PDO、プログラム可能なエラー応答、 DSP405 : NMT マスタはファンクションブロックを使用してプログラムできる (インデックス範囲の規定)
COB ID 設定	SDO、標準
Node ID 設定	DIP スイッチ
その他の CANopen 機能	NMT スレーブ、最小ブートアップ、変数 PDO マッピング、 エマージェンシメッセージ、ライフガーディング/ハートビート、仮想モジュールのコンフィグレーション
電源電圧	DC 24V (ー15%/+20%)
入力電流	最大 500mA (24V)
電源効率	87%
内部消費電力	350mA (5V)
I/O モジュール供給電流	1650mA (5V)
耐電圧	500V システム/フィールド
電源ジャンパ接点経由の電圧	DC 24V (ー15%/+30%)
電源ジャンパ接点経由の電流	最大 DC 10A
寸法 (mm) W×H×L	51×65*×100 (*取付レールの上端からの値)
重量	約 195g

4 CANopen

4.1 概要

CAN (Controller Area Network) は、自動車の電子制御装置のデータ送信用通信規格として 1980 年代中頃に開発されました。CAN 規格では物理層とデータバックアップ層であるデータリンク層を定義しています。メッセージ構造は正確に記述されていますが、アプリケーション層に関しては何も書かれていません。これに対し CAL は、アプリケーション層または送信データの意味を記述しています。CAL は CAN ネットワーク用の記述言語で、大量の通信サービスを提供します。

CANopen は、シリアルバスシステム CAN に基づいたネットワークに関する概念です。CANopen は、CiA (CAN in Automation) の DS301 規格によって一定のアプリケーション層として定義されています。

ネットワーク管理によってネットワークの立ち上げが簡単になります。このネットワークは、必要に応じユーザが拡張することができます。

CAN はマルチマスタが可能なバスシステムです。他のフィールドバスシステムに対比して、バスに接続したモジュールは、アドレス指定をせずにメッセージが識別されます。バスが空いているときはいつでも、送信元ノードはメッセージを送信することができます。バスの衝突は、メッセージに優先順位をつけることで解決しています。この優先順位は COB ID (Communication Object Identifier—通信オブジェクト識別子) によって定義され、通信オブジェクト毎に明確に割り当てられます。識別子の数字が小さいほど優先順位は高くなります。バスマスタのグループがなくても通信は可能です。

各ノードはデータ送信の時期に関しては単独で決定できます。しかしまた、他のノードがデータを送るのを要求することもできます。この要求はいわゆるリモートフレームを介して行われます。

CANopen 仕様書 (DS301) は、分散フィールド自動機器のネットワークに使用される技術上および機能上の特徴を定義しています。



詳細情報

CiA はそのメンバー用に詳細情報を次の WEB サイトに載せています。

<http://www.can-cia.de/>

4.2 ネットワークアーキテクチャ

4.2.1 送信媒体

4.2.1.1 ケーブルのタイプ

バス媒体は CAN を物理的に接続するときの基盤となります。CAN についてのバスカブリングやバス媒体は ISO11898 (CAN ハイスピード) によって規定されています。

ケーブル仕様により、特性インピーダンス $108\sim132\Omega$ のツイストペア媒体 (シールド付ツイストペアケーブル) が推奨されています。
ツイストペアは値段が安く、使用が簡単なため、単純なバス型の配線ができます。

ワゴCANopenフィールドバスノードは、シールド付銅線 ($3\times0.25\text{mm}^2$) を使用した配線をする必要があります。

バス媒体を電氣的に設計するときは、以下の重要な 2 点を考慮しなければなりません。

- 最大バス長
- 電線断面積

4.2.1.2 最大バス長

バスの長さは主に信号の伝送時間 (スピード) によって制限されますので、以下のように対応するボーレートを使用しなければなりません。

ボーレート	バス長
1Mbps	30m
800Kbps	50m
500Kbps	100m
250Kbps	250m
125Kbps	500m
$\leq 50\text{Kbps}$	1000m

表 4-1 : 設定ボーレートに対応した最大バス長

4.2.1.3 電線断面積

使用する電線の断面積は電線長により異なり、接続したノードの数に従って選択しなければなりません。

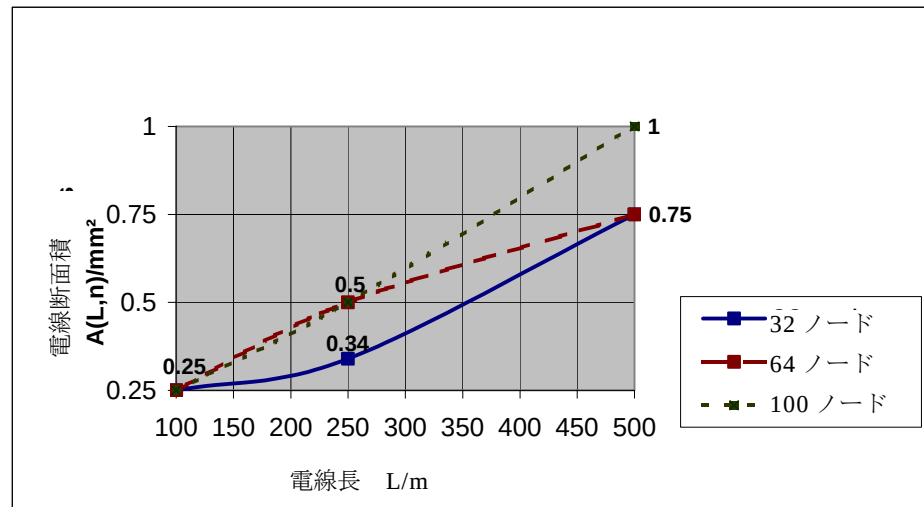


図 4-1：電線長およびノード数に従った電線断面積

4.2.2 ケーブル接続

ワゴ I/O ノードを CANopen バスケーブルに接続するには、ワゴ製の適合するプラグ（5 極 MCS プラグまたは 9 ピン D-SUB プラグ）を使用します。

シールド付銅線ケーブル（ $3 \times 0.25 \text{ mm}^2$ ）でケーブル接続する場合、プラグの各ピンにおいて CAN_High、CAN_Low および CAN_GND を結線します。

CAN_High と CAN_Low は 2 つの物理的に異なったバスレベルを表します。

CAN_GND はコモン基準電位です。

ケーブルのシールド線はドレイン端子につなげることができます。この端子はグラウンドまたは PE（キャリアレール接点）に対し $1 \text{ M}\Omega$ で終端されます。PE に対しシールドを低抵抗で接続するためには外部でのみ可能となります（電源入力モジュールの使用など）。これは全 CANopen バスケーブルのシールド線に対して、中央で PE 接続をするためです。



メモ

フィールドバス・ケーブルのシールド線と機能アースとの間で理想的な接続を実現するために、ワゴではシールド線接続システム（790 シリーズ）を用意しています。

各 CAN ノードは差動電圧 U_{Diff} を有しています。ここで $V_{\text{Diff}} = V_{\text{CAN\_High}} - V_{\text{CAN\_Low}}$ （CAN_High と CAN_Low 間のバスレベルの差）。

差動信号で送信すると、コモンモードノイズやノード間のグラウンドオフセットに対し影響されないという利点があります。

バスレベルがレセシブ状態にある場合、CAN_Low と CAN_GND 間の電圧は 2.5V になり、CAN_High と CAN_GND 間の電圧もまた 2.5V になります。つまり差動電圧は 0V になります。

バスレベルがドミナント状態にある場合、CAN_Low と CAN_GND 間の電圧は 1.5V になり、CAN_High と CAN_GND 間の電圧は 3.5V になります。このとき差動電圧は約 2V になります。



注意

ノード機器を接続するとき、信号ケーブルが交錯しないように注意してください。信号の反射を抑え送信障害を避けるために、バスの終端ではケーブルの信号線間に $120\ \Omega$ のマッチング抵抗（終端抵抗）を必ず接続しなければなりません。この抵抗はケーブルが非常に短い場合でも必要となります。

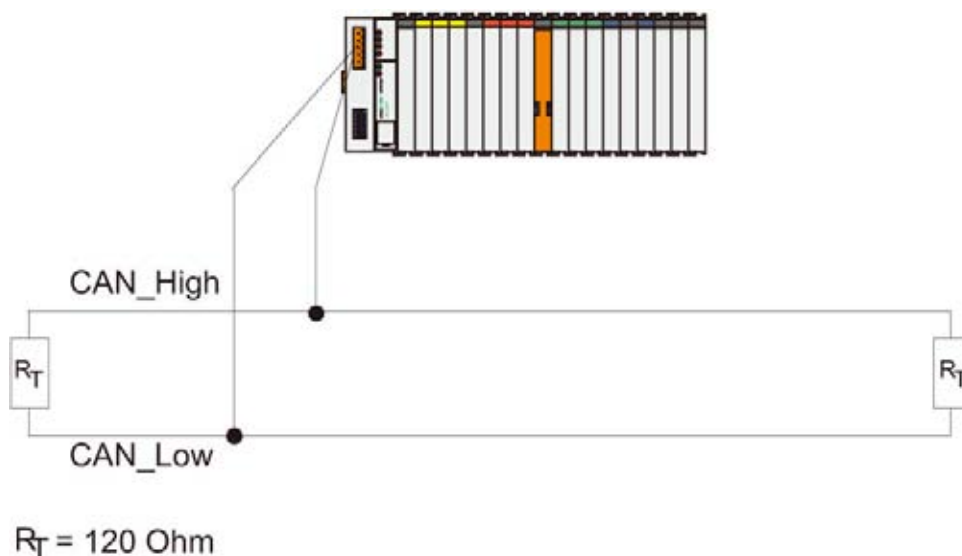


図 4-2 : ワゴ I/O ノードを CAN バスに接続する方法

g012402x

ネットワーク上でバスコントローラを立ち上げる際は、インストレーション状態を調べてください。物理的な接続が正しいかどうかは、抵抗計を CAN フィールドバスの任意の場所に置いてテストすることにより調べることができます。終端抵抗以外の全ての機器は、ネットワークからはずしてテストしなければなりません。

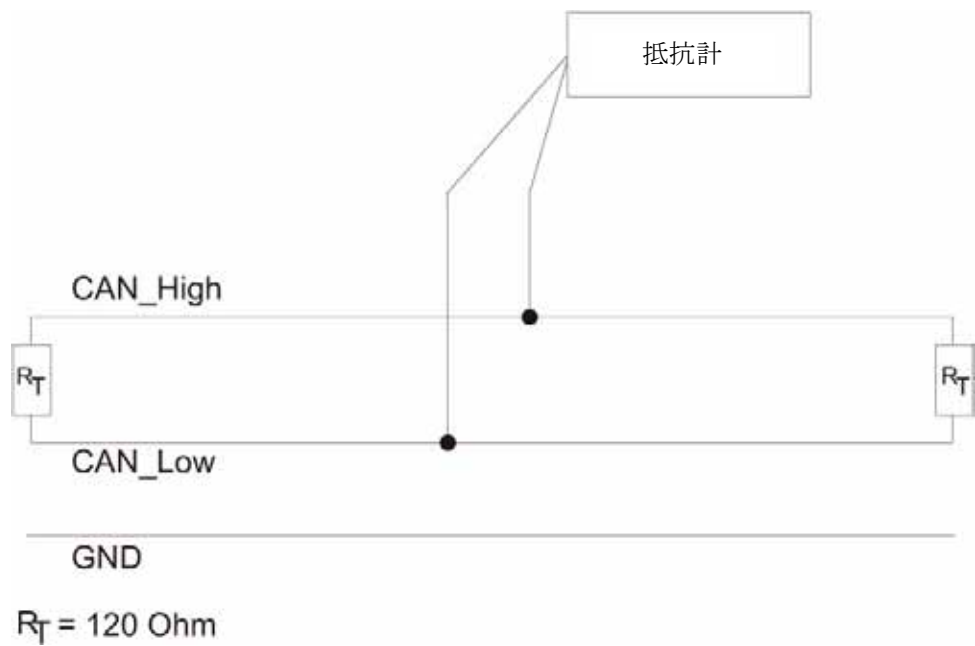


図 4-3：配線前に CAN バスをテストする際の測定方法

測定箇所	値	意味
GND と CAN_L 間	無限	OK
	0	GND と CAN_L 間で短絡
GND と CAN_H 間	無限	OK
	0	GND と CAN_H 間で短絡
CAN_L と CAN_H 間	約 60 Ω	OK、バス上に 2 個の終端抵抗あり
	約 120 Ω	バス上に 1 個の終端抵抗あり
	<50 Ω	バス上に 2 個以上の終端抵抗あり

CAN バスは 2 線式のバスであり、バスエラー管理機能を用いて、非対称動作からケーブル断または短絡を検出することができます。



詳細情報
CiA は仕様に関するドキュメント、特にケーブル仕様書を次の WEB サイトに載せています。
<http://www.can-cia.de/>

4.2.3 ネットワークトポロジ

CANopen ネットワークを簡単に構築するために、マスタ (CANopen フィールドバス PCB カード付 PC)、接続用ケーブルおよび DC24V 電源 (CANopen ノード用の電源とは別にしたもの) が必要となります。

CANopen ネットワークはマッチング抵抗 (終端抵抗: 120Ω) 付のライン構造で組まれます。

2 局以上存在するシステムでは、全てのノードは並列に配線されます。バスから各子局への接続ケーブルの最大長は 0.3m 以下にしなければなりません。

バス接続図

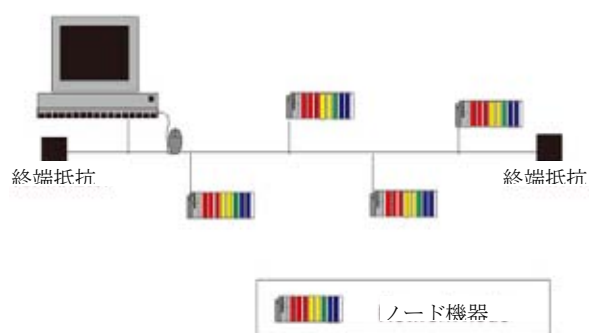


図 4-4 : CANopen ネットワークのバストポロジ

g012430e

ネットワーク上の全てのノード機器は同じボーレートで通信します。バス構造のため、他の局に影響を与えずに入出力を結合するとか、またはシステムを段階的に立ち上げることができます。

システムを後で拡張する場合、既に動作中の局に影響を与えることはありません。ノードが故障したか、またはネットワークに新たに追加されたとき、システムは自動的にこれを検出します。

ルータを使うことにより、ライン状のバスから分岐したり、ネットワークを階層的に組んだりすることが可能になります。

リピータを使うとネットワークの空間距離 (バス長) を拡張することができ、接続可能なノード数を最大 110 台まで増やすことができます。ネットワークの空間拡張距離は送信速度に依存しますが、CAN もまたネットワーク空間を拡張するのに用いることができます。達成できるデータ速度は他のバスシステムと同じ程度です。しかしケーブルの延長できる最大長は、信号遅れにより 1 リピータ当り 20~30m だけ縮められます。

4.2.4 インタフェースモジュール

ワゴ CANopen フィールドバスノードは、ネットワーク内で全てスレーブとして動作します。マスタ動作は PLC や NC などの中央制御システムが行います。フィールドバスデバイスへの接続はインタフェースモジュール経由で行われます。

プログラマブルロジックコントローラ（PLC）用のインタフェースモジュールや CANopen 用の PC インタフェース PCB は、様々なメーカーによって提供されています。

4.2.5 コンフィグレーションソフトウェア

PLC がフィールドバスデバイスと I/O データの通信をすることができるようにするために、PLC 内のマスタボードはネットワーク上の各フィールドバスデバイスに対しコンフィグレーションを行わなければなりません。

フィールドバスネットワークをコンフィグレーションし、また診断するためのソフトウェアは、PLC 用インタフェースモジュールまたは CANopen 用の PC インタフェース PCB と共に供給されるか、または CAN マスタソフトウェア（Hilscher 社の SyCon など）に含まれます。

ワゴ CANopen コントローラに必要なデータは、コンフィグレーションソフトウェアに EDS（Electrical Data Sheet）ファイルを合体することによって得ることができます。



メモ

I/O モジュールのコンフィグレーション用の EDS およびシンボルファイルは、WEB サイトから無料でダウンロードすることができます。

www.wago.co.jp/io/index.htm

4.3 ネットワーク通信

CANopen ではデータ送信、イベントのトリガ、エラー状態の通知などは通信オブジェクトを用いて行われます。このために各通信オブジェクトにはネットワーク内での COB-ID (Communication Object Identifier) が明確に割付けられています。

COB-ID の割付けはデバイスプロファイル DS401 に準じており、以下の表形式になっています。

10	9	8	7	6	5	4	3	2	1	0
機能コード				ノード ID (0 [=全部]、1~127)						

通信オブジェクト		機能コード		COB-ID		通信プロファイル 参照インデックス
		10 進	2 進	10 進	hex	hex
ブロードキャスト メッセージ	NMT コマンド	0	0000	0		
	同期メッセージ	1	0001	128	80	1005, 1006, 1007
	システム時間	2	0010	256	100	1012, 1013
	アラームオブジェクト	1	0001	129-255	81-FF	1014, 1015
ピアツーピア メッセージ	Tx-PDO1	3	0011	385-511	181-1FF	1800
	Rx-PDO1	4	0100	513-639	201-27F	1400
	Tx-PDO2	5	0101	641-767	281-2FF	1801
	Rx-PDO2	6	0110	769-895	301-37F	1401
	Tx-PDO3	7	0111	897-1023	381-3FF	1802
	Rx-PDO3	8	1000	1025-1151	401-47F	1402
	Tx-PDO4	9	1001	1153-1279	481-4FF	1803
	Rx-PDO4	10	1010	1281-1407	501-57F	1403
	Tx-SDO	11	1011	1409-1535	581-5FF	1200
	Rx-SDO	12	1100	1537-1663	601-67F	1200
	ノードモニタリング	14	1110	1793-1919	701-77F	100C, 100D 1016, 1017

通信オブジェクトに必要なパラメータ、また CANopen ノードデバイスのパラメータやデータはオブジェクトディクショナリに入っています。

ノードで利用できるオブジェクトのタイプや数は、各フィールドバスコントローラに依存します。

ネットワーク管理 (NMT) のような幾つかの特別オブジェクトに加えて、同期 (SYNC) またはエラーメッセージ (EMCY) に対しては、通信プロファイルには 2 つのオブジェクトタイプの PDO (Process Data Objects) および SDO (Service Data Objects) が存在します。PDO は実時間データ (アプリケーションデータ) の転送に用いられ、SDO はオブジェクトディクショナリにアクセスし、読み出しおよび書き込みを可能にします。

4.3.1 通信オブジェクト

4.3.1.1 PDO (Process Data Object-プロセスデータ・オブジェクト)

PDO は優先度の高い ID を持った実時間データが入っています。データ電文は最大 8 バイトで構成され、要求に応じて個々のサブアセンブリの間（プロデューサとコンシューマ間）で交換することができます。このデータ交換はイベントコントロールで任意に行うことができるか、または同期方式で実行することができます。イベントコントロールモードではバスの付加を大幅に下げることができ、低いボーレートでも大きな容量の通信が可能になります。一方、様々なモードを混合して処理することもできます (4.3.4.2.20 節「オブジェクト 0x1400 ~ 0x141F、受信 PDO 通信パラメータ」参照)。

4.3.1.1.1 PDO プロトコル

このプロトコルはバスコントローラに対し、プロトコルのオーバーヘッドなしにデータを送受信するのに用いられます。PDO は CAN 識別子とデータ領域からのみ構成されます。他のプロトコル情報は一切 PDO に含まれていません。PDO データの内容はマッピングパラメータと通信パラメータの送信タイプで定義されています。

RxPDO (受信 PDO) と TxPDO (送信 PDO) は区別して用いられます。

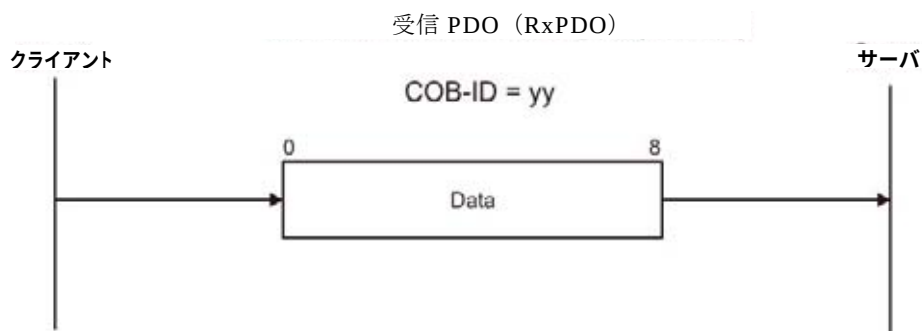


図 4-5 : RxPDO

g012403x

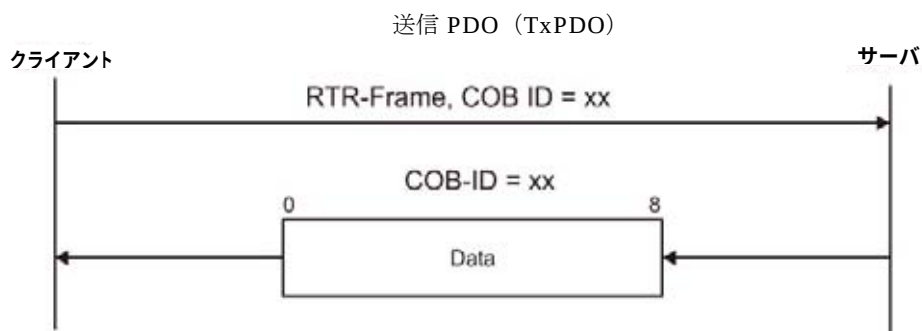


図 4-6 : TxPDO

g012404x

4.3.1.2 SDO (Service Data Object-サービスデータ・オブジェクト)

SDO はオブジェクトディクショナリの各入力値に対し、読み出しまたは書き込みをするのに用いることができます。これを利用することで CANopen ノードデバイスを完全にコンフィグレーションすることができます。

デフォルトの SDO は、優先度の低い ID で予め割り当てられています。送信するデータは 4 バイトを超えた場合、幾つかのメッセージに分けなければなりません。

4.3.1.2.1 SDO プロトコル

SDO は送信に不可欠な特定のプロトコル用オーバーヘッドを持っており、コマンド指定子、インデックスおよびリード/ライトされる入力項目のサブインデックで構成されます。

4.3.1.2.1.1 基本手順

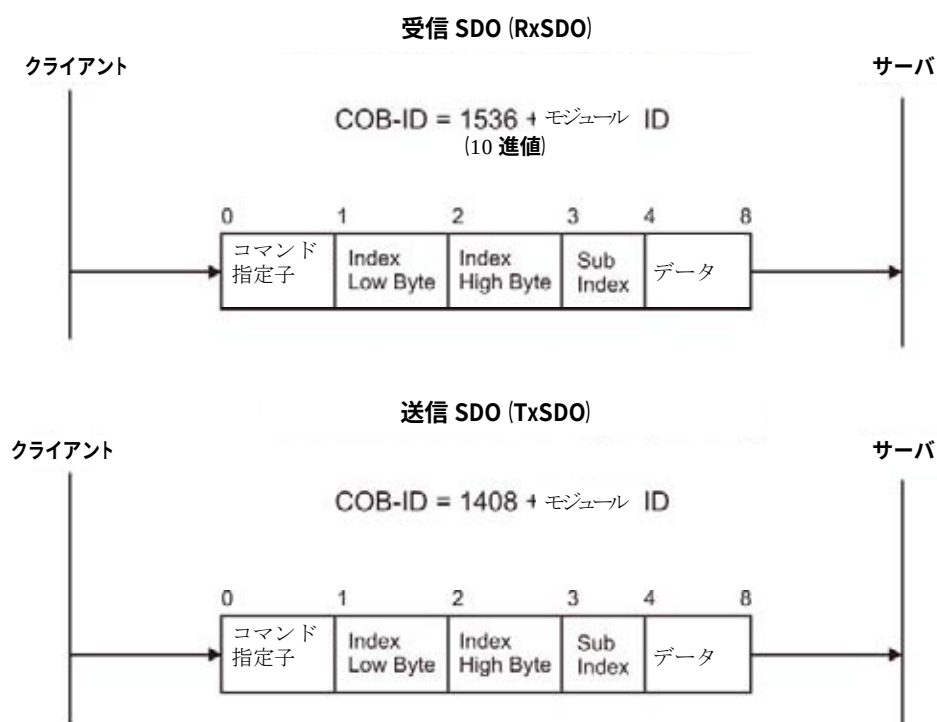


図 4-7 : SDO プロトコル

g012405x

4.3.1.2.1.2 SDO ダウンロード・プロトコル

このプロトコルは、マスタからバスコントローラにデータを書き込むときに用いられます。

4.3.1.2.1.2.1 SDO ダウンロードの開始

このプロトコルは、マスタからバスコントローラへのデータ送信を開始するときに用いられます。4 バイト以内のデータを送信するときは、データはこのプロトコルの中で送信できます。

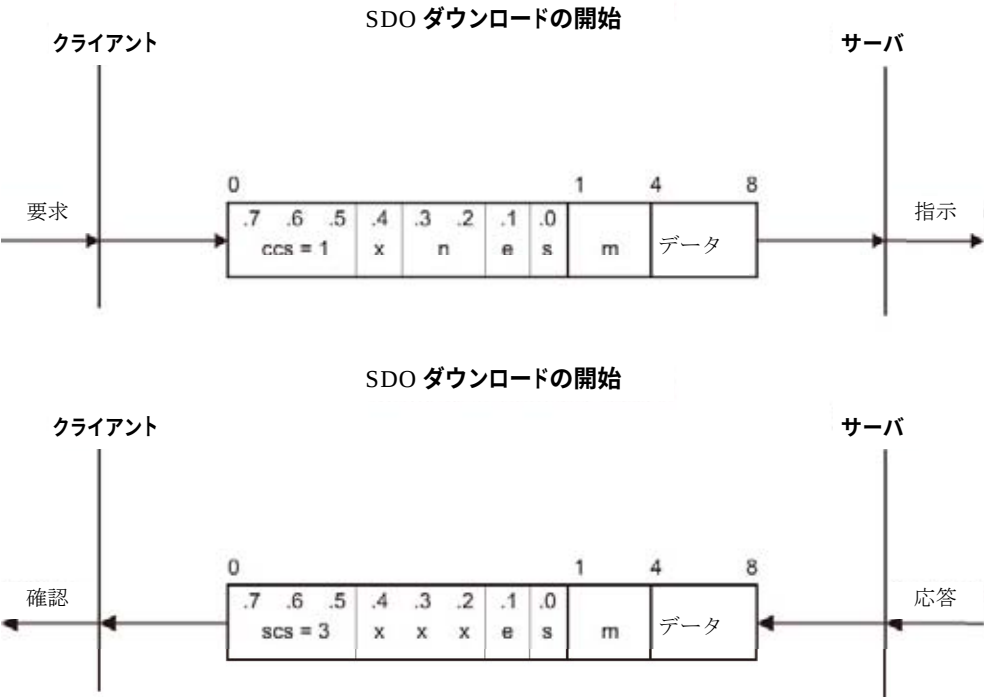


図 4-8 : SDO ダウンロードの開始

g012406x

ccs :	クライアント・コマンド指定子	1 : ダウンロード開始の要求
scs :	サーバ・コマンド指定子	3 : ダウンロード開始の応答
n :	e=1 および s=1 のときのみ有効 他の場合は 0	n が有効の場合データを含まないバイト数 を表示。例：データ=3 バイト、 e=1 および s=1 : n=4-3=1
e :	転送タイプ	0 : 標準転送、書き込むバイト数 ≥5 バイトのとき 1 : 急送。書き込むバイト数 < 5 バイトのとき
s :	サイズ指定	0 : データセットのサイズは表示しない 1 : データセットのサイズを表示する s は常に 1
m :	マルチプレクサ	オブジェクトディクショナリのインデックスとサブインデックス インデックス、下位バイト : バイト 1 インデックス、上位バイト : バイト 2 サブインデックス : バイト 3
d :	データ	e=0、s=0 : d は CiA で使用予約 e=0、s=1 : d にはダウンロードのバイト数 が入る バイト 4 は LSB、バイト 7 は MSB が入る e=1 : d にはデータが入る
x :		未使用、常に 0

4.3.1.2.1.2.2 SDO セグメントのダウンロード

このプロトコルは 4 バイト以上のデータを送信するのに用いられます。すなわち、データ送信を開始する「SDO ダウンロード開始プロトコル」を完全に処理した後に実行されます。

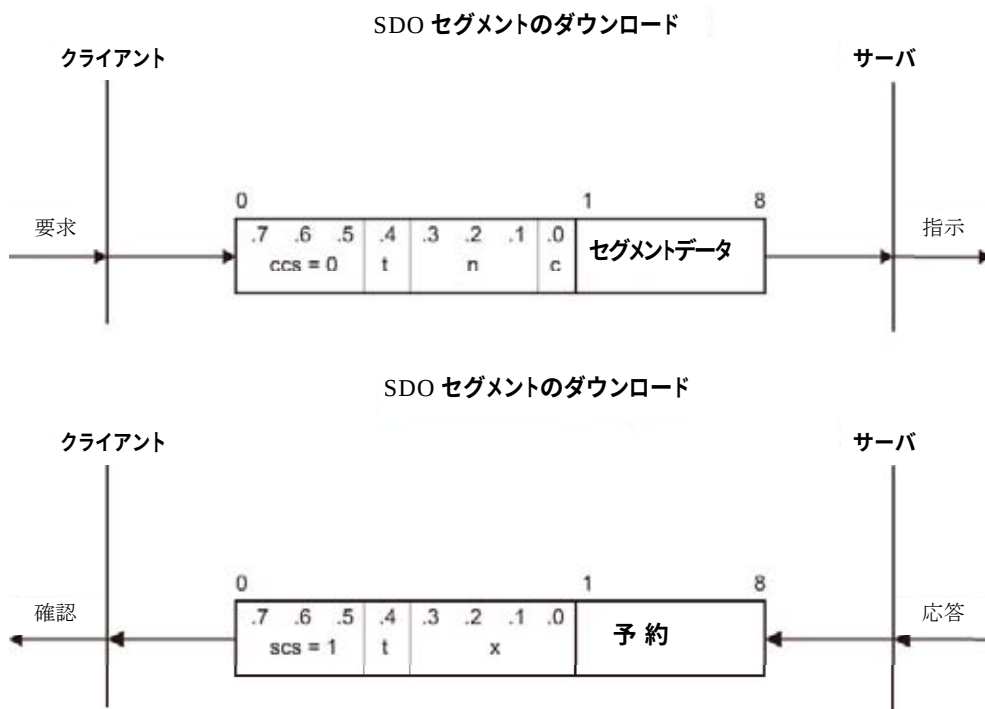


図 4-9 : SDO セグメントのダウンロード

g012409x

ccs : クライアント・コマンド指定子
scs : サーバ・コマンド指定子
セグメントデータ : 送信データが入る

n :

c : ダウンロードするデータがまだあるかどうかを示す

t : トグルビット

x :

予約 :

0 : セグメント・ダウンロードの要求

1 : セグメント・ダウンロードの応答

データの中身はアプリケーションで判断される

データを含まないバイト数を示す。

セグメント・サイズが示されない場合、n=0

0 : ダウンロードするデータがまだある

1 : ダウンロードするデータはもうない

このビットはセグメントのダウンロードが行われる毎に切り替わるものとする。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。

4.3.1.2.1.3 SDO アップロード・プロトコル

このプロトコルはバスコントローラからデータを読み出すのに用いられます。

```

sequenceDiagram
    participant Client as クライアント
    participant Server as サーバ
    Client->>Server: 要求
    Note over Client,Server: SDO アップロードの開始
    Note over Client,Server: 0 1 4 8
    Note over Client,Server: .7 .6 .5 .4 .3 .2 .1 .0
    Note over Client,Server: CCS = 2 x m 予約
    Server->>Client: 指示
  
```

SDO アップロードの開始

クライアント

サーバ

確認

応答

0 1 4 8

.7 .6 .5 .4 .3 .2 .1 .0

scs = 2 x n e s m データ

g012410x

ccs :	クライアント・コマンド指定子	2 : アップロード開始の要求
scs :	サーバ・コマンド指定子	2 : アップロード開始の応答
n :	e=1 および s=1 のときのみ有効 他の場合は 0	n が有効の場合データを含まないバイト数 を表示。バイト[8-n、7]にはセグメントデ ータはない。 e=1 および s=1、n=4-3=1 0 : 標準転送、書き込むバイト数 ≥5 バイトのとき 1 : 急送。書き込むバイト数 <5 バイトのとき
e :	転送タイプ	0 : 送信するバイト数は表示しない 1 : 送信するバイト数は表示する (バイト数に依存する)
s :	サイズ指定	オブジェクトディクショナリのインデッ クスとサブインデックス インデックス、下位バイト : バイト 1 インデックス、上位バイト : バイト 2 サブインデックス : バイト 3
m :	マルチプレクサ	e=0、s=0 : d は CiA で使用予約 c=0、s=1 : d はダウンロードのバイト数が入る バイト 4 は LSB、バイト 7 は MSB が入る e=1 : d はデータが入る
d :	データ	未使用、常に 0
x :		CiA 用に予約
予約 :		

4.3.1.2.1.3.2 SDO セグメントのアップロード

このプロトコルは 4 個以上のデータを送信するのに用いられます。すなわち、データ送信を開始する「SDO アップロード開始プロトコル」を完全に処理した後に実行されます。

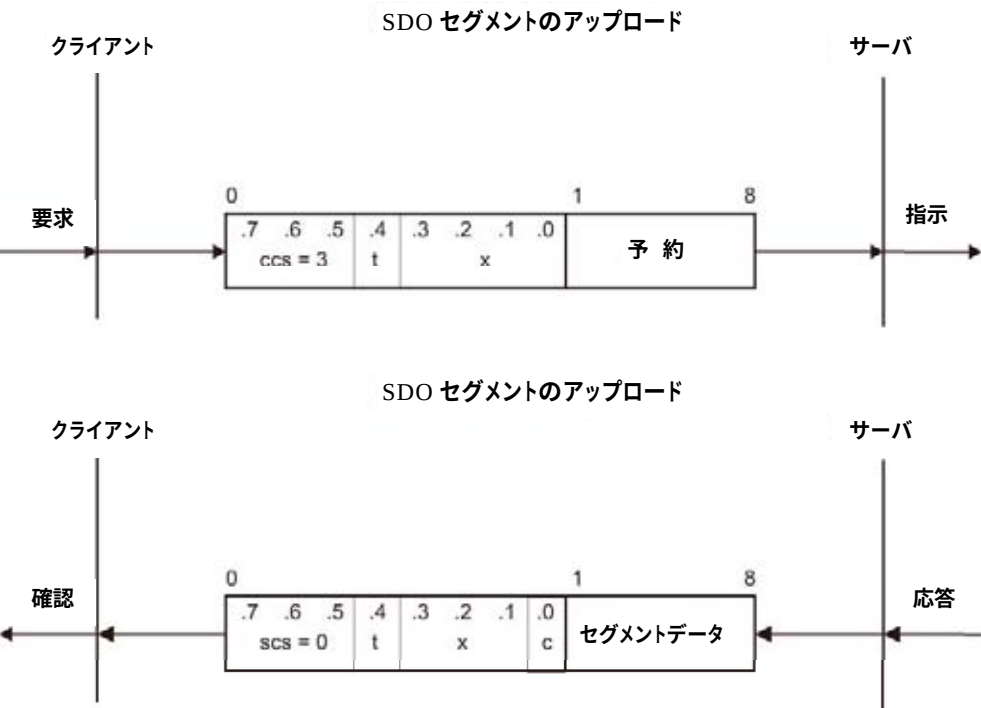


図 4-11 : SDO セグメントのアップロード g012411x

- | | | |
|------------|---------------------------|---|
| ccs : | クライアント・コマンド指定子 | 3 : セグメント・アップロードの要求 |
| scs : | サーバ・コマンド指定子 | 0 : セグメント・アップロードの応答 |
| t : | トグルビット | このビットはセグメントのアップロードが行われる毎に切り替わるものとする。最初のセグメントのトグルビットは 0 にする。トグルビットは要求と応答メッセージにおいて同じ値である。 |
| c : | アップロードするセグメントがまだあるかどうかを示す | 0 : アップロードするセグメントがまだある
1 : アップロードするデータはもうない |
| セグメントデータ : | 送信データを含む | データの中身はアプリケーションで判断される |
| n : | | データを含まないバイト数を示す。バイト[8-n, 7]にはデータはない。セグメント・サイズが示されない場合、n=0 |
| x : | | 未使用、常に 0 |
| 予約 : | | CiA により予約 |

4.3.1.2.1.4 SDO 転送のアボート

このプロトコルは送信中にエラーが起きた場合に用いられます。



図 4-12 : SDO 転送のアボート g012412x

- cs :

m :

x :

データ :
- コマンド指定子

マルチプレクサ

未使用、常に 0

4 バイト、エラーコード
- 4 : ドメイン転送のアボート

オブジェクトディクショナリのインデックスとサブインデックス

アボートの理由を示すアプリケーション固有のデータ

ドメイン転送アボートメッセージは以下のような構造になっています。

構造

バイト	意味
0	コマンド指定子
1	インデックス
2	
3	
4	追加コード
5	
6	エラーコード
7	エラークラス

エラーの中の追加コード、エラーコードおよびエラークラスは、以下の表に示すように UNSIGNED32 値としてコード化されます。

バイト 7-6 追加コード	バイト 5 エラーコード	バイト 4 エラークラス	意 味
05 03	00	00	トグルビットが切り替わらない
05 04	00	00	SDO プロトコルがタイムアウト
05 04	00	01	クライアント/サーバ・コマンド指定子が無効または不明
05 04	00	02	無効ブロック・サイズ (ブロック・モードのみ)
05 04	00	03	無効シーケンス番号 (ブロック・モードのみ)
05 04	00	04	CRC エラー (ブロック・モードのみ)
05 04	00	05	メモリ不足
06 01	00	00	オブジェクトに対するサポートされていないアクセス
06 01	00	01	ライト・オンリー・オブジェクトをリードしようとした
06 01	00	02	リード・オンリー・オブジェクトをライトしようとした
06 02	00	00	オブジェクトディクショナリにないオブジェクト
06 04	00	41	PDO にマッピングできないオブジェクト
06 04	00	42	マッピング対象オブジェクトの数および長さが PDO 長を超える
06 04	00	43	一般的なパラメータ不一致
06 04	00	47	デバイス内の一般的な内部不一致
06 06	00	00	ハードウェア・エラーによるアクセス失敗
06 07	00	10	データタイプ不一致、サービスパラメータ長の不一致
06 07	00	12	データタイプ不一致、サービスパラメータが長すぎる
06 07	00	13	データタイプ不一致、サービスパラメータが短すぎる
06 09	00	11	サブインデックスが存在しない
06 09	00	30	パラメータの値範囲を超えた (ライトアクセスのみ)
06 09	00	31	書き込まれたパラメータ値が大きすぎる
06 09	00	32	書き込まれたパラメータ値が小さすぎる
06 09	00	36	最大値が最小値より小さい
08 00	00	00	一般エラー
08 00	00	20	データをアプリケーションに転送または保存できない
08 00	00	21	ローカル制御が原因で、データをアプリケーションに転送または保存できない
08 00	00	22	現在のデバイス状態が原因で、データをアプリケーションに転送または保存できない
08 00	00	23	オブジェクトディクショナリの動的生成に失敗、またはオブジェクトディクショナリが存在しない (例えば、オブジェクトディクショナリをファイルから生成しようとしたが、ファイルエラーのために生成できなかった)

4.3.1.2.2 SDO 使用例

以下に 4 種類の SDO 使用例を掲げます。データは 16 進で表されます。この例では CAN メッセージレベルでの SDO の処理を示しており、SDO プロトコルが CAN カードに実装されていれば利用できます。

1 つのメッセージに対し、4 項目の欄に分けて説明します。

欄 1	方向	M→BC=マスタからバスコントローラにメッセージ送信 BC→M=バスコントローラからマスタにメッセージ送信
欄 2	CAN-ID	CAN 識別子
欄 3	フレームタイプ	D=データフレーム R=RTR フレーム (RTR : Remote Transmit)
欄 4	データ	CAN メッセージのデータバイト CAN メッセージでは最大 8 バイトのデータを送信できます。 個々のデータはスペースで分割されます。XX 値の項目は意味を持ちませんが、場所は確保されます。この値は'0'にしてお

いた方がよいでしょう。DD 値を持つバスコントローラからの応答項目にはコンフィグレーションに基づいたデータが入っています。

4.3.1.2.2.1 例1：インデックス 0x1000、サブインデックス 0（デバイスタイプ）のリード

インデックス 0x1000 は 4 バイトを返します。送信に急送モードが使われます。

方向	CAN-ID	フレームタイプ	データバイト 0-7
M→BC	0x601	D	0x40 00 10 00 XX XX XX XX
BC→M	0x581	D	0x43 00 10 00 91 01 DD 00

結果：

CAN-ID=0x601 は COB-ID の機能コード=1100 (Rx-SDO)、ノード ID=1 を表す
データバイト 0：40=コマンド指定子、データバイト 1-2：00 10=インデックス 0x1000、
データバイト 3：00=サブインデックス

CAN-ID=0x581 は COB-ID の機能コード=1011 (Tx-SDO)、ノード ID=1 を表す
データバイト 0：43=コマンド指定子、データバイト 1-2：00 10=インデックス 0x1000、
データバイト 3：00=サブインデックス、
データバイト 4-5： 91 01：下位バイト→高位バイトの順、ローテーションする
0x0191=401・・・デバイスプロファイル番号
データバイト 6-7： DD 00：下位バイト→高位バイトの順、ローテーションする

4.3.1.2.2.2 例2：インデックス 0x1008、サブインデックス 0（製造者デバイス名）のリード

インデックス 0x1008 は 4 バイト以上を返します。モードあたり 2 メッセージを送信する場合は標準転送モードが用いられます。

方向	CAN-ID	フレームタイプ	データバイト 0-7
M→BC	0x601	D	0x40 08 10 00 XX XX XX XX
BC→M	0x581	D	0x41 08 10 00 07 00 00 00
M→BC	0x601	D	0x60 XX XX XX XX XX XX XX
BC→M	0x581	D	0x01 37 35 30 2D 33 38 37

結果：

バスコントローラからの最初の応答は、マスタに送信するデータのバイト数を知らせます (7 バイト)。2 番目のメッセージでは、バスコントローラは ASCII フォーマット (hex 表示) で製品型番"750-837"を伝えます。

4.3.1.2.2.3 例3：インデックス 0x6000、サブインデックス 1 (8 ビット・デジタル入力ブロック-1 番目) のリード

デジタル入力モジュールの信号はインデックス 0x6000 にセーブされます。各 8 ビットが 1 グループに割り当てられ、サブインデックス 1 から読み出すことができます。この例では、8 ビットグループ 1 番目の入力値を SDO メッセージによって読み出しています。

方向	CAN-ID	フレームタイプ	データバイト 0-7
M→BC	0x601	D	0x40 00 60 01 XX XX XX XX
BC→M	0x581	D	0x4F 00 60 01 02 XX XX XX

結果：

CAN メッセージの 5 バイト目にて、バスコントローラは 8 ビットグループ 1 番目の入力値を返します。この例では 2 番目のビットがセットされています。バイト 5-7 は無効値です。

4.3.1.2.2.4 例4：インデックス 0x6200、サブインデックス 1 (8 ビット・デジタル出力ブロック-1 番目) へのライト

デジタル出力モジュールの出力値はインデックス 0x6200 にセーブされます。各 8 ビットが 1 グループに割り当てられ、サブインデックス 1 に対し読み出して、また書き込むことができます。この例では、8 ビット・デジタル出力グループ 1 番目の出力に 0xFF 値を書き込んでいます。

方向	CAN-ID	フレームタイプ	データバイト 0-7
M→BC	0x601	D	0x2F 00 62 01 FF XX XX XX
BC→M	0x581	D	0x60 00 62 01 XX XX XX XX

結果：

8 ビット・デジタル出力グループ 1 番目の出力がセットされます。

4.3.1.3 同期オブジェクト-SYNC

このオブジェクトを用いると、ネットワーク上で全てのノードの同期化が可能となります。対象の PDO を対応するようにコンフィグレーションすることにより、ネットワーク上のノードに起動をかけ、SYNC オブジェクトが到着するたびに入力データを処理するか、または出力を更新することができます。

SYNC オブジェクトを周期的に送信することにより、ネットワーク上の全てのノードがプロセスデータを同時に処理することが保証されます。

4.3.1.3.1 SYNC プロトコル

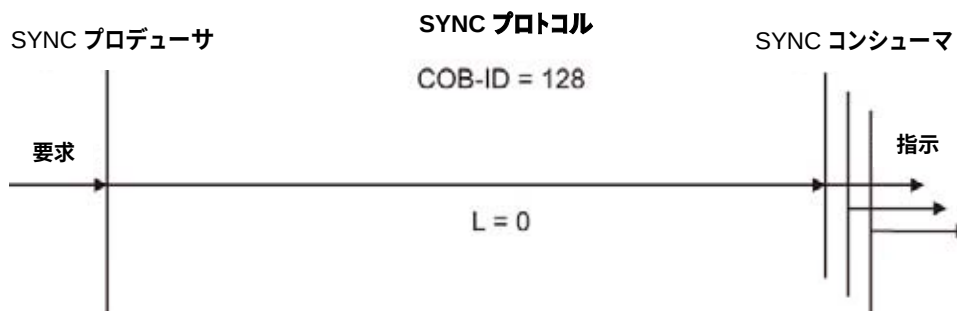


図 4-13 : SYNC プロトコル

g012413x

4.3.1.4 エマージェンシ・オブジェクト (EMCY)

エマージェンシ・オブジェクトは、内部エラーの状況によりトリガがかかります。例えば、モジュールが動作中に抜かれるとか、またはモジュールがエラー信号を通知するなどです。バスコントローラはこのとき生じたエラーを同時通報するために、接続した全てのデバイスにエマージェンシ・オブジェクトを送ります（ブロードキャスト）。このとき、通知を受けたノードはエラー修正のために適切な対策を取って応じることができます。

4.3.1.4.1 EMCY プロトコル

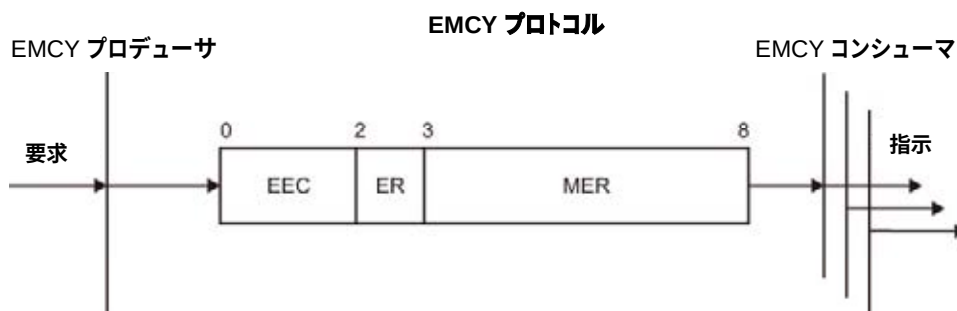


図 4-14 : EMCY プロトコル

g012414x



詳細情報

エマージェンシ・メッセージに対する詳細説明については 4.3.9 節「エラーメッセージ (エマージェンシ)」を参照してください。

4.3.2 CANopen フィールドバスコントローラの通信状態

4.3.2.1 CANopen 状態遷移図

以下に描かれた状態遷移図は、ノードの個々の通信状態と CAN 通信に関連して起こりうる遷移状態を示しています。

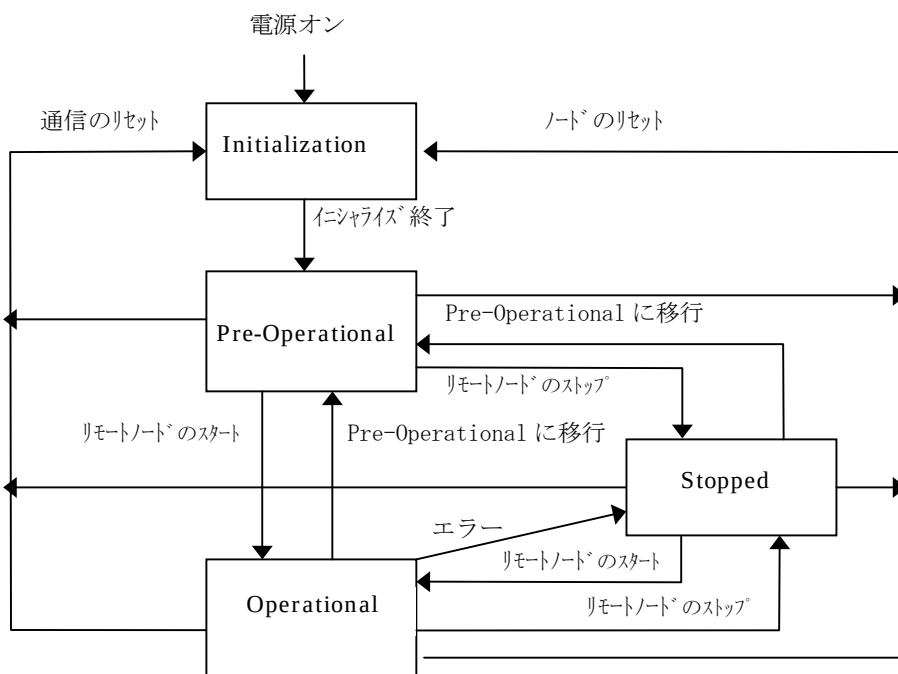


図 4-15 : フィールドバスコントローラ/コントローラの状態遷移図

4.3.2.2 INITIALIZATION

電源オンまたはリセット後バスコントローラ（モジュール ID は 0 以外）は自動的に INITIALIZATION ステータスになります。このときバスコントローラは自己チェックを行い、内部素子や通信インタフェースなどの全ての機能を調べます。実装モジュールの構成や保存されたコンフィグレーションなどを基にしてプロセスイメージが生成されます。ここでオブジェクトディクショナリがイニシャライズされます。イニシャライズ中にエラーがない場合バスコントローラは自動的に PRE-OPERATIONAL ステータスに移行します。エラーがあった場合は STOP ステータスに移ります。

イニシャライズ中 I/O LED は橙色の点滅で始まり、高速の赤点滅に変わります。イニシャライズが終わり、完全に PRE-OPERATIONAL ステータスに移行すると I/O LED は緑色になり、RUN LED が点滅します。エラーが起きた場合（例：終端モジュールなし）I/O LED は赤の点滅シーケンス（「LED 表示」を参照）によってエラータイプを示します。このとき STOP LED が赤点灯します。

4.3.2.3 PRE-OPERATIONAL

このステータスでは通信は SDO を介して行うことができます。PDO による通信はできません。

SDO によってオブジェクトディクショナリの入力項目のリード/ライトができ、コンフィグレーションツールを用いてバスコントローラをコンフィグレーションし直すことができます。マッピング、バスコントローラのパラメータ、ID などを必要条件に合わせて設定することができます。新しく作ったコンフィグレーションはフラッシュメモリに保存できます。PRE-OPERATIONAL ステータスから OPERATIONAL ステータスへの移行は NMT サービスの Start_Remote_Node コマンドによって実行します。

PRE-OPERATIONAL ステータスでは、I/O LED は緑点灯し、RUN LED が点滅します。

4.3.2.4 OPERATIONAL

このステータスでは SDO および PDO の通信が可能ですが、コンフィグレーションは同じでなければなりません。例えば有効な PDO が有る場合 COB ID の変更はできません。

OPERATIONAL ステータスから PRE-OPERATIONAL ステータスへの移行は NMT サービスの Enter_Pre_Operational_State コマンドによって行います。

OPERATIONAL ステータスでは I/O および RUN LED が点灯します。

4.3.2.5 STOPPED

STOPPED ステータスはエラー状態を表します。例としては NMT サービスの Stop_Remote_Node コマンドを受信した場合や重大な内部エラーが起きた場合（例：動作中にモジュールが抜かれた）などが挙げられます。

このステータスでは SDO および PDO の通信はできません。NMT サービスやノードゲーディング/ハートビート（アクティブ状態のとき）のみが実行できます。

NMT サービスの Start_Remote_Node_Indication、Enter_Pre_Operational_State および Reset_Node によって STOPPED ステータスを抜けることができます。

STOPPED ステータスでは STOP LED が点灯します。

4.3.3 ネットワーク管理オブジェクト

4.3.3.1 モジュール制御プロトコル

NMT（ネットワーク管理）マスタはこのプロトコルを用いてネットワーク上の全ての NMT スレーブのステータスを調べることができます。定義されたステータスには、INITIALIZING、PRE-OPERATIONAL、OPERATIONAL、STOPPED の 4 ステータスがあります。全てのノードのステータスを 1 コマンドで変えたり、また各ノードのステータスを個別に変更したりすることもできます。

4.3.3.1.1 リモートノードスタート

このサービスにより、NMT スレーブ（バスコントローラ）のステータスを OPERATIONAL に変更します。

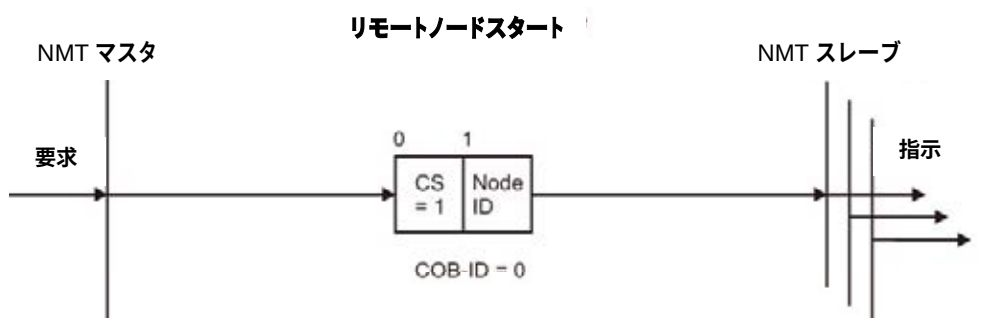


図 4-16：リモートノードスタート

g012415x

ノード ID=0：存在する全てのノードのステータスを OPERATIONAL に変更します。

4.3.3.1.2 リモートノードストップ

このサービスにより、NMT スレーブ（バスコントローラ）のステータスを STOPPED に変更します。

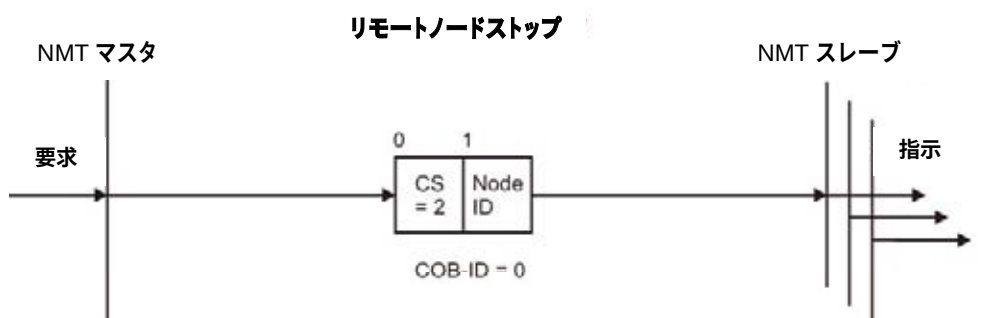


図 4-17：リモートノードストップ

g012416x

ノード ID=0：存在する全てのノードのステータスを STOPPED に変更します。

4.3.3.1.3 PRE-OPERATIONAL 移行

このサービスにより、NMT スレーブ（バスコントローラ）のステータスを PRE-OPERATIONAL に変更します。

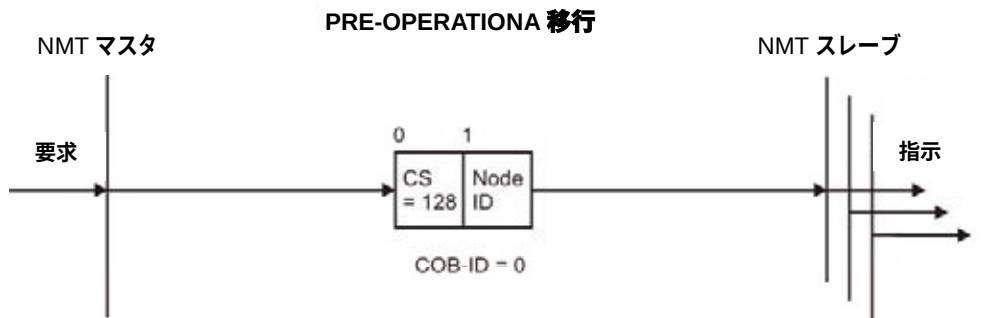


図 4-18 : PRE-OPERATIONAL 移行

g012417x

ノード ID=0 : 存在する全てのノードのステータスを PRE-OPERATIONAL に変更します。

4.3.3.1.4 ノードリセット

このサービスにより、NMT スレーブ（バスコントローラ）はリセットされます。

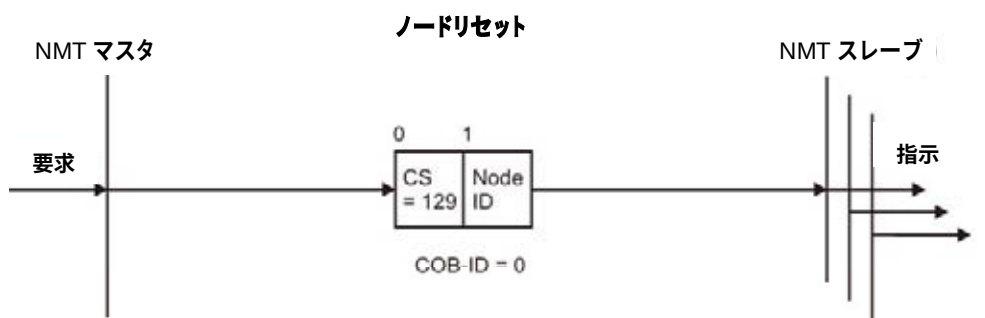


図 4-19 : ノードリセット

g012418x

ノード ID=0 : 存在する全てのノードがリセットされます。

4.3.3.2 エラー制御プロトコル

このプロトコルにより、ネットワーク上で起こりうるエラーの検出ができます。マスタはノードが定義されたステータスにまだいるかどうか、または例えばリセット後に別のステータスに移ったかを調べることができます。

4.3.3.3 ノードガーディング・プロトコル

ノードガーディングが用いられる場合、NMT スレーブはマスタからの RTR フレームによって周期的に現在の状態を送るように要求されます。付加されたトグルビットのオンオフにより、マスタは NMT スレーブがまだ正しく動作しているかどうかを検出できます。

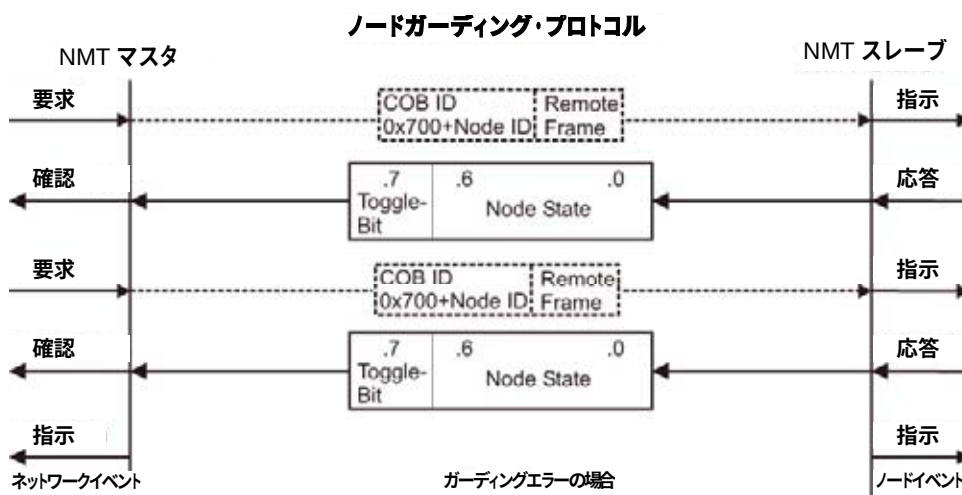


図 4-20 : ノードガーディング・プロトコル

4.3.3.4 ハートビート・プロトコル

このプロトコルを用いると、RTR フレームなしにモニタすることが可能になります。ハートビート生成ノード（プロデューサ）は周期的にハートビート・メッセージを送信し、複数の他のノード（コンシューマ）がこれを受信します。ハートビート・メッセージには現在のプロデューサの状態コードが入っています。

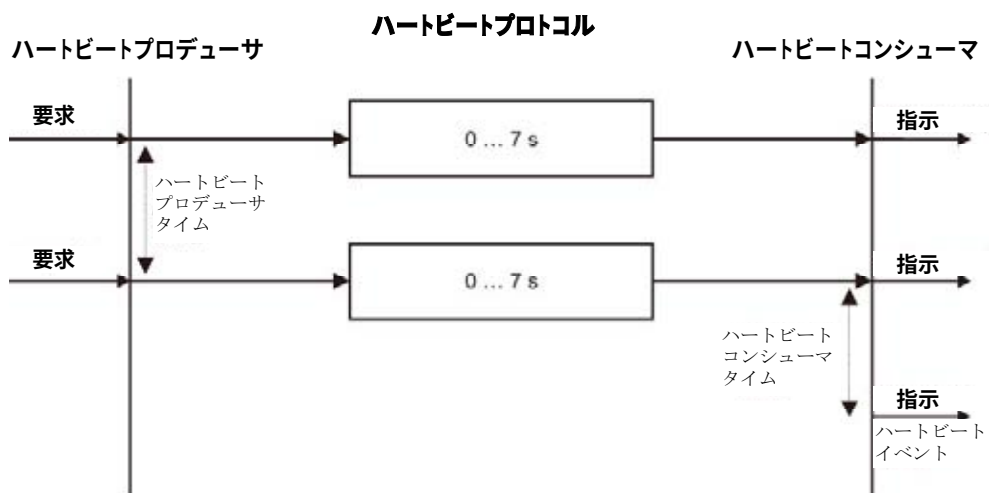


図 4-21 : ハートビート・プロトコル

g012420x

ハートビート・メッセージの中身：

- ビット 7＝ 常に 0
- ビット 6～0＝ 0 : BOOTUP
- 4 : STOPPED
- 5 : OPERATIONAL
- 127 : PRE-OPERATIONAL

4.3.3.5 ブートアップ・プロトコル

このプロトコルは NMT スレーブから、そのノード状態が INITIALIZING から PRE-OPERATIONAL に変わったことを伝えるのに用いられます。この手順はハードウェア / ソフトウェア・リセットまたはサービスリセット・コードの発行後実施されます。

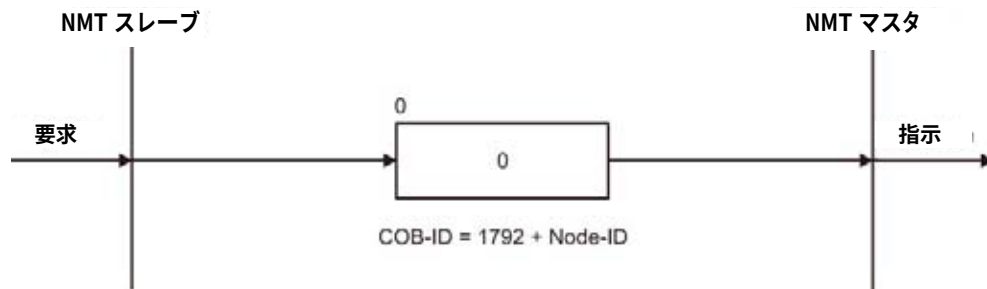


図 4-22 : ブートアップ・プロトコル

g012421x

4.3.4 オブジェクトディクショナリ

オブジェクトディクショナリは CANopen 上の各ノードの中心となるもので、コンフィグレーション情報やデータの全てが保管してあり、ポーリングで取り出すことができます。オブジェクトディクショナリはテーブル構造になっており、以下の 3 種類の CANopen オブジェクト領域が入っています。

- 通信プロファイル領域 (インデックス 0x1000 – 0x1FFF)
CANopen 通信関係のパラメータ全てが入っています。この領域は CANopen 上の全てのノードに対して同じものです。
- 製造者固有プロファイル領域 (インデックス 0x2000 – 0x5FFF)
このプロファイル領域では、各製造者は自分固有のオブジェクトを扱うことができます。
- 標準化デバイスプロファイル領域 (インデックス 0x6000 – 0x9FFF)
このプロファイルには、特定のデバイスプロファイルでサポートする全てのオブジェクトが入っています。バスコントローラはデバイスプロファイル DS-401 (汎用 I/O モジュール用デバイスプロファイル) をサポートします。

オブジェクトディクショナリでは、論理的なアドレス構造を用いて通信およびデバイスパラメータ、データ、機能などにアクセスします。オブジェクトディクショナリへの各入力値は 16bit のインデックス (テーブルの行アドレスを表す) で識別されます。最大 65536 の項目まで入れることができます。

あるオブジェクトが幾つかのモジュールから成る場合、そのモジュールは 8bit のサブインデックスによって識別されます。サブインデックスはテーブル内の個々の列アドレスを指定します (最大 256 項目まで)。

テーブルの中の各入力項目は次のような構成になっています。

- オブジェクト機能を表したオブジェクト名称
- 入力項目のデータタイプを定義したデータタイプ属性
- 入力項目がリードのみか、ライトのみか、またはリード／ライト用かどうかを示したアクセス属性

サブインデックス 0 にはこれ以降のサブインデックス入力項目の最大数（バイト数）が入ります。各サブインデックス入力項目にはコード化されたデータが入ります。

インデックス	サブインデックス	名称	タイプ	属性	デフォルト値
--------	----------	----	-----	----	--------

オブジェクトディクショナリの全体的な構造は以下のとおりです。

インデックス (hex)	オブジェクト
0x0000	未使用
0x0001 – 0x001F	スタティック（静的）データタイプ
0x0020 – 0x003F	コンプレックス（複合）データタイプ
0x0040 – 0x005F	製造者固有データタイプ
0x0060 – 0x007F	プロファイル固有スタティックデータタイプ
0x0080 – 0x009F	プロファイル固有コンプレックスデータタイプ
0x00A0 – 0x0FFF	未使用
0x1000 – 0x1FFF	通信プロファイル（DS-301）
0x2000 – 0x5FFF	製造者固有パラメータ
0x6000 – 0x9FFF	標準化デバイスプロファイルのパラメータ
0xA000 – 0xFFFF	予約領域

表 4-2 : CANopen オブジェクトディクショナリ構造

オブジェクトディクショナリは最悪のケースを想定して設計します。オブジェクト入力項目が使用できない場合は、接続モジュールのコンフィグレーションが非アクティブになっているのが原因です。

4.3.4.1 初期化 (INITIALIZATION)

実装したモジュールのコンフィグレーションは電源投入後に決定されます。

ユーザ固有のコンフィグレーションがセーブされており、現在実装されたモジュールのコンフィグレーションが最後にセーブしたものと一致したならば、このセーブしたコンフィグレーションのオブジェクトディクショナリが初期化されます。

これ以外の全ての場合では、オブジェクトディクショナリはデフォルトのコンフィグレーションが割り当てられます。

4.3.4.1.1 デフォルト・コンフィグレーション

4.3.4.1.1.1 通信プロファイル領域の初期化

バスコントローラでサポートされているプロファイルの全てのオブジェクトは、DS301 (CANopen アプリケーション層および通信プロファイル) のデフォルト値に基づいて初期化されます。

- デフォルト・マッピングパラメータの入力
マッピングパラメータの初期設定はデバイスプロファイルに依存します。バスコントローラは DS401 をサポートしており、そこに記述されたプロセスが用いられます。最初の 4 個の Rx/TxPDO はデフォルト PDO として定義されます。入出力点数がこれ以上の場合は、5 番目の Rx/TxPDO 以降に残りの I/O を割り当てます。最初にデジタル、その後アナログ I/O を入れます。入出力に対し 64 点以上のデジタル I/O があった場合 (アナログモジュールがなくても)、5 番目 PDO から続けることができます。このとき PDO2~4 は不利用のままとなります。更に 1 つのデータタイプのみが PDO のデフォルト入力として用いられます。つまり 3 バイトまたは 4 バイトタイプのモジュールが存在した場合は 2 個の PDO 各々に対してデフォルト値を入力します。

第 1 RxPDO :

最大 8 個の 8 点デジタル出力が入ります。デジタル出力がなければサブインデックス 0 は '0' となり、この PDO に対しデフォルト値は用いられません。

以降の表においては Idx=インデックス、S-Idx=サブインデックスを表します。

Idx	S-Idx	内容	デフォルト値
0x1600	0	マッピングされるオブジェクト数	0 : デジタル出力ブロックなし 1~8 : 1~8 個のデジタル出力ブロック
	1	マッピングされる 1 番目のデジタル出力ブロック	0x6200 01 08
	2	マッピングされる 2 番目のデジタル出力ブロック	0x6200 02 08
	...	...	...
	8	マッピングされる 8 番目のデジタル出力ブロック	0x6200 08 08

第 2 RxPDO :

1～4 番目以内の 16bit アナログ出力が入ります。アナログ出力がなければサブインデックス 0 は'0'となり、この PDO に対しデフォルト値は用いられません。

Idx	S-Idx	内容	デフォルト値
0x1601	0	マッピングされるオブジェクト数	0 : アナログ出力なし 1～4: 1～4 個のアナログ出力
	1	マッピングされる 1 番目のアナログ出力	0x6411 01 10
	2	マッピングされる 2 番目のアナログ出力	0x6411 02 10
	3	マッピングされる 3 番目のアナログ出力	0x6411 03 10
	4	マッピングされる 4 番目のアナログ出力	0x6411 04 10

第 3 RxPDO :

5～8 番目以内の 16bit アナログ出力が入ります。4 点以上のアナログ出力がなければサブインデックス 0 は'0'となり、この PDO に対しデフォルト値は用いられません。

Idx	S-Idx	内容	デフォルト値
0x1602	0	マッピングされるオブジェクト数	0 : アナログ出力なし 1～4: 1～4 個のアナログ出力
	1	マッピングされる 5 番目のアナログ出力	0x6411 05 10
	2	マッピングされる 6 番目のアナログ出力	0x6411 06 10
	3	マッピングされる 7 番目のアナログ出力	0x6411 07 10
	4	マッピングされる 8 番目のアナログ出力	0x6411 08 10

第 4 RxPDO :

9～12 番目以内の 16bit アナログ出力が入ります。8 点以上のアナログ出力がなければサブインデックス 0 は'0'となり、この PDO に対しデフォルト値は用いられません。

Idx	S-Idx	内容	デフォルト値
0x1603	0	マッピングされるオブジェクト数	0 : アナログ出力なし 1～4: 1～4 個のアナログ出力
	1	マッピングされる 9 番目のアナログ出力	0x6411 09 10
	2	マッピングされる 10 番目のアナログ出力	0x6411 0A 10
	3	マッピングされる 11 番目のアナログ出力	0x6411 0B 10
	4	マッピングされる 12 番目のアナログ出力	0x6411 0C 10

第 1 TxPDO :

最大 8 個の 8 点デジタル入力が入ります。デジタル入力がなければサブインデックス 0 は'0'となり、この PDO に対しデフォルト値は用いられません。

Idx	S-Idx	内容	デフォルト値
0x1A00	0	マッピングされるオブジェクト数	0 : デジタル入力ブロックなし 1～8: 1～8 個のデジタル入力ブロック
	1	マッピングされる 1 番目のデジタル入力ブロック	0x6000 01 08
	2	マッピングされる 2 番目のデジタル入力ブロック	0x6000 02 08
	...	...	...
	8	マッピングされる 8 番目のデジタル入力ブロック	0x6000 08 08

第 2 TxPDO :

1～4 番目以内の 16bit アナログ入力が入ります。アナログ入力が無ければサブインデックス 0 は'0'となり、この PDO に対しデフォルト値は用いられません。

Idx	S-Idx	内容	デフォルト値
0x1A01	0	マッピングされるオブジェクト数	0 : アナログ入力なし 1～4: 1～4 個のアナログ入力
	1	マッピングされる 1 番目のアナログ入力	0x6401 01 10
	2	マッピングされる 2 番目のアナログ入力	0x6401 02 10
	3	マッピングされる 3 番目のアナログ入力	0x6401 03 10
	4	マッピングされる 4 番目のアナログ入力	0x6401 04 10

第 3 TxPDO :

5～8 番目以内の 16bit アナログ入力が入ります。4 点以上のアナログ出力が無ければサブインデックス 0 は'0'となり、この PDO に対しデフォルト値は用いられません。

Idx	S-Idx	内容	デフォルト値
0x1A02	0	マッピングされるオブジェクト数	0 : アナログ入力なし 1～4: 1～4 個のアナログ入力
	1	マッピングされる 5 番目のアナログ入力	0x6401 05 10
	2	マッピングされる 6 番目のアナログ入力	0x6401 06 10
	3	マッピングされる 7 番目のアナログ入力	0x6401 07 10
	4	マッピングされる 8 番目のアナログ入力	0x6401 08 10

第 4 TxPDO :

9～12 番目以内の 16bit アナログ入力が入ります。8 点以上のアナログ入力が無ければサブインデックス 0 は'0'となり、この PDO に対しデフォルト値は用いられません。

Idx	S-Idx	内容	デフォルト値
0x1A03	0	マッピングされるオブジェクト数	0 : アナログ入力なし 1～4: 1～4 個のアナログ入力
	1	マッピングされる 9 番目のアナログ入力	0x6401 09 10
	2	マッピングされる 10 番目のアナログ入力	0x6401 0A 10
	3	マッピングされる 11 番目のアナログ入力	0x6401 0B 10
	4	マッピングされる 12 番目のアナログ入力	0x6401 0C 10

製造者固有プロファイル領域の初期化

この領域はオブジェクトディクショナリに記述されているように初期化されます。

標準化デバイスプロファイル領域の初期化

DS401 規格に定義されているように、サポートされた全てのオブジェクトが初期化されます。

4.3.4.2 通信プロファイル領域

以下の表には、バスコントローラでサポートされている全ての通信プロファイル・オブジェクトが記載されています。

インデックス	名称	タイプ	意味	参照頁
0x1000	デバイスタイプ	Unsigned32	デバイスプロファイル	111
0x1001	エラーレジスタ	Unsigned8	ビット単位でエラーを定義 (DS401)	111
0x1003	定義済みエラーフィールド	Array Unsigned32	発生エラーのうち最近 20 件を保存	112
0x1005	COB-ID SYNC メッセージ	Unsigned32	SYNC オブジェクトの COB-ID	112
0x1006	通信周期	Unsigned32	2 つの SYNC メッセージ間の最大時間	112
0x1008	製造者デバイス名	可視文字列	製造者のデバイス名	112
0x1009	製造者ハードウェアバージョン	可視文字列	製造者のハードウェアバージョン	112
0x100A	製造者ソフトウェアバージョン	可視文字列	製造者のソフトウェアバージョン	113
0x100C	ガードタイム	Unsigned16	ライフガーディング・プロトコルに使用する時間	113
0x100D	ライフタイムファクタ	Unsigned8	寿命ファクタ、ノードガーディング・プロトコルの一部	113
0x1010	パラメータ保存	Array Unsigned32	コンフィグレーション用パラメータの保存	113
0x1011	デフォルトパラメータ復元	Array Unsigned32	コンフィグレーションのデフォルトパラメータの復元	114
0x1014	COB-ID エマージェンシ・オブジェクト	Unsigned32	エマージェンシ・オブジェクトの COB-ID	115
0x1015	EMCY Inhibit タイム	Unsigned32	2 つの EMCY メッセージ間の最小時間	115
0x1016	コンシューマ・ハートビート時間	Array Unsigned32	ハートビートをモニタリングする時間	115
0x1017	プロデューサ・ハートビート時間	Unsigned16	2 つのハートビート・メッセージ間の時間	116
0x1018	アイデンティティ・オブジェクト	レコード アイデンティティ	デバイスの一般的情報	116
0x1200 - 0x1201	サーバ SDO パラメータ	レコード SDO パラメータ	サーバ SDO のパラメータ	116
0x1280 - 0x128F	クライアント SDO パラメータ	レコード SDO パラメータ	クライアント SDO のパラメータ	117
0x1400 - 0x141F	受信 PDO 通信パラメータ	レコード PDO パラメータ	受信 PDO の通信パラメータ	117
0x1600 - 0x161F	受信 PDO マッピングパラメータ	レコード PDO マッピング	受信 PDO のマッピングパラメータ	118
0x1800 - 0x181F	送信 PDO 通信パラメータ	レコード PDO マッピング	送信 PDO の通信パラメータ	118
0x1A00 - 0x1A1F	送信 PDO マッピングパラメータ	レコード PDO マッピング	送信 PDO のマッピングパラメータ	119

以降の章において表中の略号は以下の意味を表します。

Idx=インデックス

S-Idx=サブインデックス

RO=リードオンリー

RW=リードアンドライト

4.3.4.2.1 オブジェクト 0x1000、デバイスタイプ

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1000	0	デバイスタイプ	Unsigned32	RO	-

このオブジェクトは実装されたデバイスプロファイルを表します。CANopen バスコントローラには「汎用 I/O モジュール用デバイスプロファイル (デバイスプロファイル 401)」が実装されています。また、インデックス 0x1000 には接続したモジュールのタイプ情報が入っています。

構造

MSB		LSB	
0000.0000	0000.4321	デバイスプロファイル番号 0x01 (上位バイト)	デバイスプロファイル番号 0x91 (下位バイト)

ビット 1=1 のとき、最低 1 個のデジタル入力接続されている
 ビット 2=1 のとき、最低 1 個のデジタル出力接続されている
 ビット 3=1 のとき、最低 1 個のアナログ入力接続されている
 ビット 4=1 のとき、最低 1 個のアナログ出力接続されている

4.3.4.2.2 オブジェクト 0x1001、エラーレジスタ

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1001	0	エラーレジスタ	Unsigned8	RO	-

このレジスタには内部エラーが入ります。このレジスタはまたエマージェンシ・メッセージの一部を表します。

構造

ビット	意味
0	一般エラー
1	電流
2	電圧
3	温度
4	通信
5	デバイスプロファイル固有
6	予約
7	製造者固有

エラーの場合、ビット 0 が常にセットされます。その他のビットにより詳細のエラーを規定しています。

4.3.4.2.3 オブジェクト 0x1003、定義済みエラーフィールド

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1003	0	エラー数	Unsigned8	RW	0
	1	標準エラーフィールド	Unsigned32	RO	-
	:	:	:	:	:
	20	標準エラーフィールド	Unsigned32	RO	-

サブインデックス 0 には、以降のフィールドに現在保存されているエラー数が入ります。新規のエラーが起きた場合、それはサブインデックス 1 に入り、既存のエラーは全て 1 つづつ次のサブインデックスに移動します。最大 20 までのエラー入力をサポートします。20 以上のエラーが起きた場合は、その度にサブインデックス 20 に入っているエラーが上書きされます。

標準エラーフィールドの構造

ビット 31	ビット 16	ビット 15	ビット 0
追加情報			エラーコード

追加情報はエマージェンシ・テレグラムの中で最初の 2 バイトの追加コードに相当します。エラーコードはエマージェンシ・テレグラムのエラーコードに一致します。エラーメモリ全体はサブインデックス 0 に"0"を書き込むことにより消去されます。

4.3.4.2.4 オブジェクト 0x1005, COB-ID SYNC メッセージ

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1005	0	COB-ID SYNC	Unsigned32	RW	0x00000080

このオブジェクトには同期メッセージの COB ID が定義されます。

構造

ビット 31	ビット 11	ビット 10	ビット 0
予約（常に 0）			COB-ID

4.3.4.2.5 オブジェクト 0x1006、通信周期

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1006	0	通信周期	Unsigned32	RW	0

このオブジェクトには2つの連続した SYNC メッセージ間の最大時間が μs で定義されます。内部の分解能は 2ms です。値が 0 の場合 SYNC の監視は実行されません。

4.3.4.2.6 オブジェクト 0x1008、製造者デバイス名

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1008	0	製造者デバイス名	可視文字列	RO	750-837 (750-838)

このオブジェクトにはバスコントローラのデバイス名が入ります。

4.3.4.2.7 オブジェクト 0x1009、製造者ハードウェアバージョン

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1009	0	製造者ハードウェアバージョン	可視文字列	RO	現在のハードウェアバージョン

このオブジェクトにはバスコントローラの現在のハードウェアバージョンが入ります。

4.3.4.2.8 オブジェクト 0x100A 製造者ソフトウェアバージョン

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x100A	0	製造者ソフトウェアバージョン	可視文字列	RO	現在のソフトウェアバージョン

このオブジェクトにはバスコントローラの現在のソフトウェアバージョンが入ります。

4.3.4.2.9 オブジェクト 0x100C、ガードタイム

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x100C	0	ガードタイム	Unsigned16	RW	0

このオブジェクトにはガードタイムがミリ秒単位で入ります。NMT マスタは NMT スレーブを周期的に問い合わせ、ステータスを取得します。2 つの問い合わせ間の時間をガードタイムと呼びます。

4.3.4.2.10 オブジェクト 0x100D、ライフタイムファクタ

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x100D	0	ライフタイムファクタ	Unsigned8	RW	0

ライフタイムファクタはノードガードプロトコルの一部です。NMT スレーブはノードライフタイム（ガードタイム×ライフタイムファクタ）以内に問い合わせを受けたかどうかチェックします。もし受けなかった場合、NMT マスタがもう正常に動作していないという前提でスレーブが動きます。このときスレーブは「ライフガードイベント」をトリガします。ノードのライフタイムが 0 の場合モニタリングは行われません。

4.3.4.2.11 オブジェクト 0x1010、パラメータ保存

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1010	0	サポートされるサブインデックスの最大値	Unsigned8	RO	1
	1	全パラメータ保存	Unsigned32	RW	1

このオブジェクトの使用により、ユーザが設定した内容を永久に保存することが可能になります。これを実現するためには署名文字の「save (ASCII 小文字、0x65 76 61 73... MSB から LSB の順)」をインデックス 0x1010、サブインデックス 1 に書き込まなければなりません。保存処理はバックグラウンドで行われ、約 2～3 秒かかります。保存処理が終了すると SDO 応答テレグラムが送信されます。SDO を用いることにより、通信は引き続き保存中に可能です。以前の保存がまだ終わっていないときに新たに保存しようすると、エラーが通知されます。「復元」がアクティブになっている間は保存機能をトリガすることはできません。パラメータ設定が保存され次第、「HW コンフィグレーションが変更された」というエマージェンシメッセージは、バスコントローラがモジュールのコンフィグレーション変更なしに再度立ち上がった場合は送信されません。



注意

コンフィグレーションを保存した後 DIP スイッチによりモジュール ID のみを変更された場合、保存されたコンフィグレーションが継続して用いられます。すなわちオブジェクトディクショナリ内のモジュール ID 固有の入力項目全て（モジュール ID に依存したオブジェクトで”rw”属性を持っている）は従来の値で応答します。

4.3.4.2.12 オブジェクト 0x1011 デフォルトパラメータの復元

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1011	0	サポートされるサブインデックスの最大値	Unsigned8	RO	4
	1	全パラメータをデフォルト値で設定	Unsigned32	RW	1
	2	-	Unsigned32	RW	0
	3	-	Unsigned32	RW	0
	4	全パラメータを1回のみデフォルト値で設定	Unsigned32	RW	1

このオブジェクトを使用すると、ユーザが保存したパラメータを元のデフォルト値に戻すことができます。

サブインデックス 2 および 3 はサポートしていません。

ロード命令はバックグラウンドで行われ、約 2～3 秒かかります。実行が終了したとき、SDO 応答メッセージが送信されます。通信は、SDO を用いることにより実行中に続けることができます。以前のロードがまだ終了していない場合、更にロード命令を送ろうとしたときのみエラーメッセージが通知されます。また、“Save”がアクティブの間はロード命令をトリガすることはできません。

4.3.4.2.12.1 サブインデックス 1—デフォルトパラメータの恒久入力

署名文字“load”（ASCII 小文字—0x64 0x61 0x66 0x6C・・・MSB～LSB の順）をインデックス 0x1011、サブインデックス 1 に書き込むと、次に電源をオンした後やその後の電源オンをする度に、標準ファクトリ設定のローディングをすることになります（次の SAVE 命令が発行されるまで）。

4.3.4.2.12.2 サブインデックス 4—デフォルトパラメータの入力オン—オフ

署名文字“load”（ASCII 小文字—0x64 0x61 0x6F 0x6C・・・MSB～LSB の順）をインデックス 0x1011、サブインデックス 4 に書き込むと、次に電源をオンした後 1 回だけ標準ファクトリ設定のローディングをすることになります。セーブされたコンフィグレーションはその後電源オンをする毎に再ロードされます。この機能は例えば開発期間中に用いられ、セーブしたコンフィグレーションとデフォルトのものを全てのパラメータを毎回再設定、再復元することなく、素早く動作の比較を得ることができるようになります。

復元手順：

デフォルト・コンフィグレーションで1回だけコンフィグレーションを置き換える。

→ load（インデックス 0x1011、サブインデックス 4）

→ リセット

→ デフォルト値

（load（インデックス 0x1011、サブインデックス 4）命令の再利用はこのステータスでは許されません。

→ リセット

→ コンフィグレーション保存

4.3.4.2.13 オブジェクト 0x1014、COB-ID エマージェンシ・オブジェクト

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1014	0	COB ID EMCY	Unsigned32	RW	0x80+モジュール ID

このオブジェクトにはエマージェンシ・メッセージ（EMCY）の COB ID が定義されます。

構造

ビット 31	ビット 30	ビット 11	ビット 10	ビット 0
0/1 有効/無効	予約 (常に 0)	COB-ID		

新規の COB ID を入力する場合、ビット 31 を最初にセットしてください。これは標準 DS301 では COB ID が有効なとき（ビット 31=0）の変更を許していないからです。

4.3.4.2.14 オブジェクト 0x1015、エマージェンシ・Inhibit タイムオブジェクト

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1015	0	EMCY Inhibit タイム	Unsigned16	RW	0

このオブジェクトでは、更にエマージェンシを送信する前に経過すべき時間が分単位で表されます。

0 を入力すると遅延送信の動作を停止します。

遅延送信のために入力項目がキューに入力されるという事実により、速く連続するエマージェンシの最大数はキューのサイズ（20 入力）に制限されます。この数字を超えた場合、エマージェンシはオーバーフローを示して直ちに送信されます。

時間単位は 100μs です。

例： 2 つの EMCY 間の最小時間間隔は 30ms です。
インデックス 0x1015=300=0x12C

4.3.4.2.15 オブジェクト 0x1016、コンシューマ・ハートビート時間

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1016	0	モニタ可能モジュールの最大数	Unsigned8	RO	5
	1	1 番目ハートビート時間入力	Unsigned32	RW	0
	2	2 番目ハートビート時間入力	Unsigned32	RW	0
	3	3 番目ハートビート時間入力	Unsigned32	RW	0
	4	4 番目ハートビート時間入力	Unsigned32	RW	0
	5	5 番目ハートビート時間入力	Unsigned32	RW	0

この入力により、最大 5 モジュールのモニタリングが可能になります。システムは、このオブジェクトで定義された各モジュールが、設定時間内にハートビートを生成したかどうかをチェックします。設定時間を超えた場合ハートビート・イベントがトリガされます。ハートビート時間はミリ秒で入力します。時間値が 0 の場合モニタリングはストップします。

構造

	MSB	LSB
ビット	31-24	23-16
値	予約	モジュール ID
データタイプ	-	Unsigned8
		Unsigned16

4.3.4.2.16 オブジェクト 0x1017、プロデューサ・ハートビート時間

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1017	0	プロデューサ・ハートビート時間	Unsigned16	RW	0

このオブジェクトには2つのハートビート・メッセージ間の時間がミリ秒で定義されます。時間値が0の場合ハートビートは送出されません。0以外の値が入力され次第ハートビート送信が開始します。

4.3.4.2.17 オブジェクト 0x1018、アイデンティティ・オブジェクト

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1018	0	サポートされる入力項目の最大数	Unsigned8	RO	4
	1	製造者 ID	Unsigned32	RO	33
	2	デバイス記述	Unsigned32	RO	337
	3	リビジョン番号	Unsigned32	RO	実リビジョン番号
	4	シリアル番号	Unsigned32	RO	実シリアル番号

このオブジェクトは使用デバイスを規定しています。
製造者 ID には、個々の製造者に割り当てられた明白な番号が入っています。ワゴは ID33 が割り当てられています。
デバイス記述は製品ファミリを表しています。
リビジョン番号には特定の CANopen ビヘイビアが入り、メジャー・リビジョン番号とマイナー・リビジョン番号で構成されます。メジャー・リビジョン番号には CANopen 機能が入ります。機能に変更された場合メジャー・リビジョン番号は増加します。CANopen ビヘイビアが同一のもので、バージョンが各種ある場合はマイナー・リビジョン番号で区別することができます。

リビジョン番号の構造

ビット 31	ビット 16	ビット 15	ビット 0
メジャー・リビジョン番号		マイナー・リビジョン番号	

シリアル番号はこのデバイス・ファミリの中で明白に識別されます。

4.3.4.2.18 オブジェクト 0x1200– 0x1201、サーバ SDO

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1200～0x1201	0	サポートされる入力項目の最大数	Unsigned8	RO	2
	1	COB-ID クライアント→サーバ (Rx)	Unsigned32	Idx 0x1200 RO Idx 0x1201 RW	Idx 0x1200 0x600+モジュール ID Idx 0x1201 0x80000000
	2	COB-ID サーバ→クライアント (Tx)	Unsigned32	Idx 0x1200 RO Idx 0x1201 RW	Idx 0x1200 0x580+モジュール ID Idx 0x1201 0x80000000

オブジェクトディクショナリの入力項目へのアクセスはこのオブジェクトを介しておこなわれます。

2 番目の SDO のデフォルト値はアクティブではありません。COB ID の変更は、現在の SDO がアクティブの場合（ビット 31=0）2 番目 SDO では禁止されます。

COB-IDの構造

ビット 31	ビット 30	ビット 11	ビット 10	ビット 0
0/1 有効/無効		予約（常に 0）		COB-ID

4.3.4.2.19 オブジェクト 0x1280– 0x128F、クライアント SDO

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1280～ 0x128F	0	サポートされる入力項目の最大数	Unsigned8	RO	3
	1	COB-ID クライアント→サーバ (Tx)	Unsigned32	RW	0x80000000
	2	COB-ID サーバ→クライアント (Rx)	Unsigned32	RW	0x80000000
	3	SDO サーバのモジュール ID	Unsigned8	RW	0

このオブジェクトにより、コントローラ付のフィールドバスノードのマスタ動作が可能になります。

フィールドバスコントローラは現在 16 個のクライアント SDO をサポートします。

フィールドバスノードは 1 個のクライアント SDO だけ処理できます。この処理が終了したとき、更にクライアント SDO を送ることができます。ネットワーク管理マスタの機能により、コントローラは最大 20 モジュールを管理することができます。

COB-IDの構造

ビット 31	ビット 30	ビット 11	ビット 10	ビット 0
0/1 有効/無効	予約 (常に 0)			COB-ID

4.3.4.2.20 オブジェクト 0x1400– 0x141F、受信 PDO 通信パラメータ

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1400～ 0x141F	0	サポートされる入力項目の最大数	Unsigned8	RO	2
	1	COB-ID	Unsigned32	RW	Idx 0x1400 0x200+モジュール ID Idx 0x1401 0x300+モジュール ID Idx 0x1402 0x400+モジュール ID Idx 0x1403 0x500+モジュール ID Idx 0x1404-141F 0x80000000
	2	送信タイプ	Unsigned8	RW	255

このオブジェクトは RxPDO の通信パラメータをセットするのに用いられます。32 個の RxPDO がサポートされます。DS301 規格に則り、最初の 4 個の PDO にはデフォルトの COB ID が予め割り当てられています。これ以降の全ての PDO は無効になっています。デフォルトの PDO の全部が必ずしも使用されない場合 (モジュールの接続数が少ないときなど) もまた、その不使用 PDO は無効となります。

COB-IDの構造

ビット 31	ビット 30	ビット 29	ビット 11	ビット 10	ビット 0
0/1 有効/無効	0/1 RTR 許可/ 非許可	予約 (常に 0)			COB-ID

新規の COB ID を入力する場合は、最初にビット 31 を 1 にセットしなければなりません。それは DS301 規格では、有効 COB ID (ビット 31=0) の変更は許可していないからです。データ送信 (インデックス通信パラメータ中の送信タイプ) のために、各 PDO に対してモードを定義することができます。標準ではデジタルおよびアナログ入力は "Change of Value(COV)" として送信されます。送信タイプによって定義される PDO 送信の種類は以下のテーブルに説明します。

送信タイプ	PDO 送信						
	周期的	非周期的	同期	非同期	RTR 時のみ	TxPDO (入力)	RxPDO (出力)
0		X	X			SYNC 毎に COV が送信される	最後に受信した PDO の要求に応じ各 SYNC の後出力をセット
1 – 240	X		X			X 個の SYNC 毎に送信される (x = 1 to 240)	最後に受信した PDO の要求に応じ各 SYNC の後出力をセット
241 – 251	-予約 -						
252			X		X	データが再度 SYNC で読み込まれる、送信はされない RTR 経由の要求	サポートされない
253				X	X	RTR 経由の要求	COV
254				X		COV ¹	COV
255				X		COV ¹	COV

¹ データは設定した Inhibit タイムの間隔で送信される

4.3.4.2.21 オブジェクト 0x1600– 0x161F、受信 PDO マッピングパラメータ

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1600～0x161F	0	マッピングされたオブジェクト数	Unsigned8	RW	-
	1～8	1 番目のオブジェクト～8 番目のオブジェクト	Unsigned32	RW	-

このオブジェクトは PDO によって送信するデータを定義するために用いられます。サブインデックス 0 には PDO として有効なオブジェクト数が入ります。

1 番目から 8 番目のオブジェクトの構造

ビット 31	ビット 16	ビット 15	ビット 8	ビット 7	ビット 0
インデックス		サブインデックス		サイズ	

インデックス 送信されるオブジェクトのインデックス (16 ビット)

サブインデックス 送信されるオブジェクトのサブインデックス (8 ビット)

サイズ オブジェクト長 (ビット単位)
1 個の PDO では最大 8 バイトが送信できるので、有効なオブジェクト長の合計は 64 (8 バイト×8 ビット) を超えてはいけません。

4.3.4.2.22 オブジェクト 0x1800– 0x181F、送信 PDO 通信パラメータ

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1800～0x181F	0	サポートされる入力項目の最大数	Unsigned8	RO	5
	1	COB-ID	Unsigned32	RW	Idx 0x1800 0x180+モジュール ID Idx 0x1801 0x280+モジュール ID Idx 0x1802 0x380+モジュール ID Idx 0x1803 0x480h+モジュール ID Idx 0x1804-181F 0x80000000
	2	送信タイプ	Unsigned8	RW	255
	3	Inhibit タイム	Unsigned16	RW	Idx 0x1800 0 Idx 0x1801 – 181F 100
	4	予約	Unsigned8	RW	0
	5	イベントタイマ	Unsigned16	RW	0

このオブジェクトは TxPDO の通信パラメータをセットするのに用いられます。

32 個の TxPDO がサポートされます。DS301 規格に則り、最初の 4 個の PDO にはデフォルトの COB ID が予め割り当てられています。これ以降の全ての PDO は無効になっています。デフォルトの PDO の全部が必ずしも使用されない場合（モジュールの接続数が少ないときなど）もまた、その不使用 PDO は無効となります。

Inhibit タイムは、同じ COB ID を持った 2 つの連続した PDO 間の最小時間（この間は送信不可能）を表します。時間単位は 100us です。送信時間の値はミリ秒単位に内部で桁下げを行います。

新規の COB ID を入力する場合は、最初にビット 31 を 1 にセットしなければなりません。それは DS301 規格では、有効 COB ID (ビット 31=0) の変更は許可していないからです。

例 同じ COB ID を持った 2 つの連続した PDO 間の最小時間間隔: 30ms
サブインデックス 3=300=0x12C

イベントタイマには、ある PDO が送信されてから次の PDO の送信までの経過時間（PDO データの変更がなかったとしても）を定義します。時間はミリ秒で入れます。このタイマはイベントが起き次第再スタートします（PDO データが変化したとき）。

タイマが Inhibit タイムより短い場合、次のイベントは Inhibit タイムが経過した後に発生します。



注意

イベントタイマは送信タイプ 254、255 でのみ使用できます。



注意

オブジェクトの入力項目は最大 3 種類の異なった PDO にのみマッピングすることができます。

4.3.4.2.23 オブジェクト 0x1A00 – 0x1A1F、送信 PDO マッピングパラメータ

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x1A00～ 0x1A1F	0	マッピングされたオブジェクト数	Unsigned8	RW	-
	1～8	1 番目のオブジェクト～ 8 番目のオブジェクト	Unsigned32	RW	-

このオブジェクトは PDO によって送信するデータを定義するために用いられます。

サブインデックス 0 には PDO として有効なオブジェクト数が入ります。

1 番目から 8 番目のオブジェクトの構造

ビット 31	ビット 16	ビット 15	ビット 8	ビット 7	ビット 0
インデックス		サブインデックス		サイズ	

インデックス 送信されるオブジェクトのインデックス
 (16 ビット)

サブインデックス 送信されるオブジェクトのサブインデックス
(8ビット)

サイズ オブジェクト長（ビット単位）
1 個の PDO では最大 8 バイトが送信できるので、有効なオブジェクト長の合計は 64（8 バイト×8 ビット）を超えてはいけません。

4.3.4.3 製造者固有プロファイル領域

非標準デバイスプロファイル固有の特殊モジュールやその他モジュールの I/O 機能、また特殊機能（スペーサモジュール・コンフィグレーションなど）は「製造者固有プロファイル領域」に描かれます。そこで定義されるオブジェクトは 1～8 バイトのデータ幅を持ったモジュールに対するワードデータ長を提供します。

インデックス 0x2000（デジタル入力）、0x2100（デジタル出力）、0x2400（2 バイト特殊モジュール入力）および 0x2500（2 バイト特殊モジュール出力）はデバイスプロファイル DS 401（0x6000、0x6200、0x6401、0x6411）の対応するインデックスによりミラーコピーされます。これは例えばオブジェクト 0x2000 とオブジェクト 0x6000 はプロセスイメージにおいて同じメモリを参照することを意味します。

以下の表には、バスコントローラでサポートしている製造者プロファイルの全てのオブジェクトを示します。

Idx	名称	タイプ	意味	参照頁
0x2000	デジタル入力	Array Unsigned8	デジタル入力モジュールのデータ	121
0x2100	デジタル出力	Array Unsigned8	デジタル出力モジュールのデータ	121
0x2200	1 バイト特殊モジュール、入力	Array Unsigned8	1 バイト特殊入力モジュールのデータ	121
0x2300	1 バイト特殊モジュール、出力	Array Unsigned8	1 バイト特殊出力モジュールのデータ	121
0x2400	2 バイト特殊モジュール、入力	Array Unsigned16	2 バイト特殊入力モジュールのデータ	121
0x2500	2 バイト特殊モジュール、出力	Array Unsigned16	2 バイト特殊出力モジュールのデータ	121
0x2600	3 バイト特殊モジュール、入力	Record	3 バイト特殊入力モジュールのデータ	121
0x2700	3 バイト特殊モジュール、出力	Record	3 バイト特殊出力モジュールのデータ	122
0x2800	4 バイト特殊モジュール、入力	Record	4 バイト特殊入力モジュールのデータ	122
0x2900	4 バイト特殊モジュール、出力	Record	4 バイト特殊出力モジュールのデータ	122
0x3000	5 バイト特殊モジュール、入力	Record	5 バイト特殊入力モジュールのデータ	122
0x3100	5 バイト特殊モジュール、出力	Record	5 バイト特殊出力モジュールのデータ	122
0x3200	6 バイト特殊モジュール、入力	Record	6 バイト特殊入力モジュールのデータ	122
0x3300	6 バイト特殊モジュール、出力	Record	6 バイト特殊出力モジュールのデータ	122
0x3400	7 バイト特殊モジュール、入力	Record	7 バイト特殊入力モジュールのデータ	123
0x3500	7 バイト特殊モジュール、出力	Record	7 バイト特殊出力モジュールのデータ	123
0x3600	8 バイト特殊モジュール、入力	Record	8 バイト特殊入力モジュールのデータ	123
0x3700	8 バイト特殊モジュール、出力	Record	8 バイト特殊出力モジュールのデータ	123
0x4200-0x4202	ゲートウェイモジュール、入力	Record	ゲートウェイモジュールの入力データ	123
0x4300-0x4302	ゲートウェイモジュール、出力	Record	ゲートウェイモジュールの出力データ	124
0x4500	スペーサモジュール・コンフィグレーション	Record	仮想モジュールのコンフィグレーション	124
0x5000	入力 PA	Record	入力プロセスイメージの読み出し	129
0x5001	出力 PA	Record	出力プロセスイメージへの書き込み	129
0x5200	コントローラ・コンフィグレーションオブジェクト	Record	PDO 処理の停止、点滅コード表示、I/O モジュール診断	129
0x5201	診断コンフィグレーションオブジェクト	Array of Byte	I/O モジュールの診断ビヘイビア	130
0x5202	モジュール・コンフィグレーションオブジェクト	Array of Byte	実装 I/O モジュールの物理的コンフィグレーション	130

4.3.4.3.1 オブジェクト 0x2000、デジタル入力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x2000	0	8ビットデジタル入力ブロック	Unsigned8	RO	-	デジタル8ビット入力ブロック数
	1	1 番目入力ブロック	Unsigned8	RO	-	1 番目デジタル入力ブロック
						
	32	32 番目入力ブロック	Unsigned8	RO	-	32 番目デジタル入力ブロック

4.3.4.3.2 オブジェクト 0x2100、デジタル出力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x2100	0	8ビットデジタル出力ブロック	Unsigned8	RO	-	デジタル8ビット出力ブロック数
	1	1 番目出力ブロック	Unsigned8	RW	0	1 番目デジタル出力ブロック
						
	32	32 番目出力ブロック	Unsigned8	RW	0	32 番目デジタル出力ブロック

4.3.4.3.3 オブジェクト 0x2200、1 バイト特殊モジュール、入力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x2200	0	特殊 1 バイト入力	Unsigned8	RO	-	1 バイト特殊チャンネル数
	1	1 番目特殊入力	Unsigned8	RO	-	1 番目入力チャンネル
						
	254	254 番目特殊入力	Unsigned8	RO	-	254 番目入力チャンネル

4.3.4.3.4 オブジェクト 0x2300、1 バイト特殊モジュール、出力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x2300	0	特殊 1 バイト出力	Unsigned8	RO	-	1 バイト特殊チャンネル数
	1	1 番目特殊出力	Unsigned8	RW	0	1 番目出力チャンネル
						
	254	254 番目特殊出力	Unsigned8	RW	0	254 番目出力チャンネル

4.3.4.3.5 オブジェクト 0x2400、2 バイト特殊モジュール、入力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x2400	0	特殊 2 バイト入力	Unsigned8	RO	-	2 バイト特殊チャンネル数
	1	1 番目特殊入力	Unsigned16	RO	-	1 番目入力チャンネル
						
	254	254 番目特殊入力	Unsigned16	RO	-	254 番目入力チャンネル

4.3.4.3.6 オブジェクト 0x2500、2 バイト特殊モジュール、出力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x2500	0	特殊 2 バイト出力	Unsigned8	RO	-	2 バイト特殊チャンネル数
	1	1 番目特殊出力	Unsigned16	RW	0	1 番目出力チャンネル
						
	254	254 番目特殊出力	Unsigned16	RW	0	254 番目出力チャンネル

4.3.4.3.7 オブジェクト 0x2600、3 バイト特殊モジュール、入力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x2600	0	特殊 3 バイト入力	Unsigned8	RO	-	3 バイト特殊チャンネル数
	1	1 番目特殊入力	Unsigned24	RO	-	1 番目入力チャンネル
						
	170	170 番目特殊入力	Unsigned24	RO	-	170 番目入力チャンネル

4.3.4.3.8 オブジェクト 0x2700、3 バイト特殊モジュール、出力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x2700	0	特殊 3 バイト出力	Unsigned8	RO	-	3 バイト特殊チャンネル数
	1	1 番目特殊出力	Unsigned24	RW	0	1 番目出力チャンネル
						
	170	170 番目特殊出力	Unsigned24	RW	0	170 番目出力チャンネル

4.3.4.3.9 オブジェクト 0x2800、4 バイト特殊モジュール、入力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x2800	0	特殊 4 バイト入力	Unsigned8	RO	-	4 バイト特殊チャンネル数
	1	1 番目特殊入力	Unsigned32	RO	-	1 番目入力チャンネル
						
	128	128 番目特殊入力	Unsigned32	RO	-	128 番目入力チャンネル

4.3.4.3.10 オブジェクト 0x2900、4 バイト特殊モジュール、出力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x2900	0	特殊 4 バイト出力	Unsigned8	RO	-	4 バイト特殊チャンネル数
	1	1 番目特殊出力	Unsigned32	RW	0	1 番目出力チャンネル
						
	128	128 番目特殊出力	Unsigned32	RW	0	128 番目出力チャンネル

4.3.4.3.11 オブジェクト 0x3000、5 バイト特殊モジュール、入力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x3000	0	特殊 5 バイト入力	Unsigned8	RO	-	5 バイト特殊チャンネル数
	1	1 番目特殊入力	Unsigned40	RO	-	1 番目入力チャンネル
						
	102	102 番目特殊入力	Unsigned40	RO	-	102 番目入力チャンネル

4.3.4.3.12 オブジェクト 0x3100、5 バイト特殊モジュール、出力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x3100	0	特殊 5 バイト出力	Unsigned8	RO	-	5 バイト特殊チャンネル数
	1	1 番目特殊出力	Unsigned40	RW	0	1 番目出力チャンネル
						
	102	102 番目特殊出力	Unsigned40	RW	0	102 番目出力チャンネル

4.3.4.3.13 オブジェクト 0x3200、6 バイト特殊モジュール、入力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x3200	0	特殊 6 バイト入力	Unsigned8	RO	-	6 バイト特殊チャンネル数
	1	1 番目特殊入力	Unsigned48	RO	-	1 番目入力チャンネル
						
	85	85 番目特殊入力	Unsigned48	RO	-	85 番目入力チャンネル

4.3.4.3.14 オブジェクト 0x3300、6 バイト特殊モジュール、出力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x3300	0	特殊 6 バイト出力	Unsigned8	RO	-	6 バイト特殊チャンネル数
	1	1 番目特殊出力	Unsigned48	RW	0	1 番目出力チャンネル
						
	85	85 番目特殊出力	Unsigned48	RW	0	85 番目出力チャンネル

4.3.4.3.15 オブジェクト 0x3400、7 バイト特殊モジュール、入力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x3400	0	特殊 7 バイト入力	Unsigned8	RO	-	7 バイト特殊チャンネル数
	1	1 番目特殊入力	Unsigned56	RO	-	1 番目入力チャンネル
						
	73	73 番目特殊入力	Unsigned56	RO	-	73 番目入力チャンネル

4.3.4.3.16 オブジェクト 0x3500、7 バイト特殊モジュール、出力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x3500	0	特殊 7 バイト出力	Unsigned8	RO	-	7 バイト特殊チャンネル数
	1	1 番目特殊出力	Unsigned56	RW	0	1 番目出力チャンネル
						
	73	73 番目特殊出力	Unsigned56	RW	0	73 番目出力チャンネル

4.3.4.3.17 オブジェクト 0x3600、8 バイト特殊モジュール、入力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x3600	0	特殊 8 バイト入力	Unsigned8	RO	-	8 バイト特殊チャンネル数
	1	1 番目特殊入力	Unsigned64	RO	-	1 番目入力チャンネル
						
	64	64 番目特殊入力	Unsigned64	RO	-	64 番目入力チャンネル

4.3.4.3.18 オブジェクト 0x3700、8 バイト特殊モジュール、出力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x3700	0	特殊 8 バイト出力	Unsigned8	RO	-	8 バイト特殊チャンネル数
	1	1 番目特殊出力	Unsigned64	RW	0	1 番目出力チャンネル
						
	64	64 番目特殊出力	Unsigned64	RW	0	64 番目出力チャンネル

4.3.4.3.19 オブジェクト 0x4200-0x4202、ゲートウェイモジュール入力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x4200-0x4202	0	使用サブインデックス最大値	Unsigned8	RO	-	サポートされるサブインデックスの最大値
	1	メールボックス長	Unsigned8	RO	-	メールボックスのサイズ
	2	メールボックス	Octet String	RO	-	メールボックス
	3-30	ゲートウェイプロセスデータ	Unsigned8	RW	-	ゲートウェイモジュールのプロセスデータ

各モジュールに対して 1 つのステータスバイトがマッピングされます。これはメールボックスの 1 番最初にあります。この後に空白バイト、実際のメールボックスおよびデータが続きます。

例：

ステータス=1 バイト、メールボックス= 6 バイト、データ= 4 バイト

入力プロセスイメージのデータ											
S	-	MB1	MB2	MB3	MB4	MB5	MB6	D1	D2	D3	D4

オブジェクトディクショナリの入力項目	
Sub0	6
Sub1	8
Sub2	S - MB1 MB2 MB3 MB4 MB5 MB6
Sub3	D1
Sub4	D2
Sub5	D3
Sub6	D4

4.3.4.3.20 オブジェクト 0x4300-0x4302 ゲートウェイモジュール出力

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x4300-0x4302	0	使用サブインデックス最大値	Unsigned8	RO	-	サポートされるサブインデックスの最大値
	1	メールボックス長	Unsigned8	RO	-	メールボックスのサイズ
	2	メールボックス	Octet String	RW	-	メールボックス
	3-30	ゲートウェイプロセスデータ	Unsigned8	RW	-	ゲートウェイモジュールのプロセスデータ

各モジュールに対して 1 つのコントロールバイトがマッピングされます。これはメールボックスの 1 番最初にあります。この後に空白バイト、実際のメールボックスおよびデータが続きます。

例：

コントロール=1 バイト、メールボックス= 6 バイト、データ= 4 バイト

出力プロセスイメージのデータ											
C	-	MB1	MB2	MB3	MB4	MB5	MB6	D1	D2	D3	D4

オブジェクトディクショナリの入力項目									
Sub0	6								
Sub1	8								
Sub2	C	-	MB1	MB2	MB3	MB4	MB5	MB6	
Sub3	D1								
Sub4	D2								
Sub5	D3								
Sub6	D4								

4.3.4.3.21 オブジェクト 0x4500 スペーサモジュール・コンフィグレーション

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x4500	0	配置および仮想 I/O モジュール数	Unsigned8	RW	0	0：不使用 1.. 64: 配置 I/O モジュールの最大数 (実装+仮想)
	1	1 番目 I/O モジュール記述	Unsigned16	RW	0	1 番目 I/O モジュール
						
	64	64 番目 I/O モジュール記述	Unsigned16	RW	0	64 番目 I/O モジュール

このオブジェクトを利用することにより、ノード内に仮想モジュールを挿入することができます。例えば拡張するノードの最大値を設計したり、最大コンフィグレーションに対しての一部分となる新規のノードを続けて設計したりすることができます。実装モジュールに対応するオブジェクト入力項目に関する新ノードのビヘイビアは、最大コンフィグレーションのビヘイビアと同じです。従って、他のアプリケーション (CANopen マスタなど) は最大コンフィグレーションに対して 1 回だけ設計でき、その設定を変更する必要がなく、どんな一部分にもアクセスできます。

同様に設計過程の最初から将来の拡張に対し備えることができますので、マッピングを後で実行することは不要になります。

- サブインデックス 0
サブインデックス 0=0：仮想モジュールのマッピングは不使用
サブインデックス 0≠0：仮想モジュールのマッピングは使用

入力値は最大コンフィグレーションにおける実装モジュールの数を示します。
値が 0 から>0 に変わったとき、コンフィグレーションはサブインデックス 1 に記述されているように生成されます。新しいコンフィグレーションの生成中、永久に保存されていない全ての以前のコンフィグレーション設定は上書きされ、プロセスイメージはリセットされます。従って、まずインデックス 0x4500 を常に最初にコンフィグレーションし、それから他の全ての設定（マッピング、同期時間など）に移ります。

サブインデックス 0 の設定は PRE-OPERATIONAL ステータスでのみ可能です。

新しいコンフィグレーションの生成にエラーがなければ、エマージェンシ・メッセージがパラメータ PP=LL=SS=0 で送信されます。

新しいコンフィグレーションの生成中にエラーが起きた場合（実装モジュール数がコンフィグレーションしたモジュール数を超えた場合など）、対応するエマージェンシ・メッセージが送信されます。コントローラは実装モジュールに基づいたデフォルトコンフィグレーションで立ち上がり、STOP ステータスに移行します。

- エマージェンシ・メッセージ
エラーコード 0x5000
エラーレジスタ 0x81
- 追加コード 00 03 PP LL SS
PP：エラーが起きた場所の実際のモジュールスロットを示します。
LL：エラーが起きた場所のロジック上のスロット（最大コンフィグレーションのスロット）を示します。
SS：エラーの原因

コンフィグレーション生成エラーが起きた場合のエマージェンシ・メッセージ構造

パラメータ: SS (エラーの原因)		
ビット 4...7	ビット 0...3	説明
0	1	コンフィグレーションに則ったアナログモジュール
0	2	コンフィグレーションに則ったデジタルモジュール
0	3	コンフィグレーションに則った出力モジュール
0	4	コンフィグレーションに則った入力モジュール
0 N	5	デジタルモジュール - 全モジュールに対しビット数が誤り (ビット数/チャンネル×チャンネル数) アナログモジュール - コンフィグレーションに則った n チャンネルのモジュール
0	6	実装モジュール数がコンフィグレーションでの数を超えた
0	7	デジタルモジュール：無効 アナログモジュール：チャンネル当りのバイト数の誤り ゲートウェイモジュール：プロセスイメージの全体サイズ誤り
0	8	コンフィグレーションに則ったゲートウェイモジュール
0	9	メールボックスサイズの指示誤り
0	10	実装モジュール数がコンフィグレーションでの数以下だった



メモ

モジュール診断メッセージがエマージェンシ・メッセージによって発生した場合、モジュール位置の表示は常にノード内のロジック上のモジュール位置を表します。従ってこのメッセージはノードコンフィグレーションにかかわらず常に同一です。

● サブインデックス 1～64

サブインデックス 1～64 にはノードコンフィグレーションが最大のコンフィグレーションで入ります。各インデックスは実装モジュールに対応します（サブインデックス 1=1 番目のモジュール、サブインデックス 2=2 番目のモジュール、・・・）。以下のインデックスでは対応するモジュールの詳細を表します。

サブインデックスの構造

MSB										LSB						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
M	予約			ビット数/バイト数						チャンネル数			出力	入力	A/D	

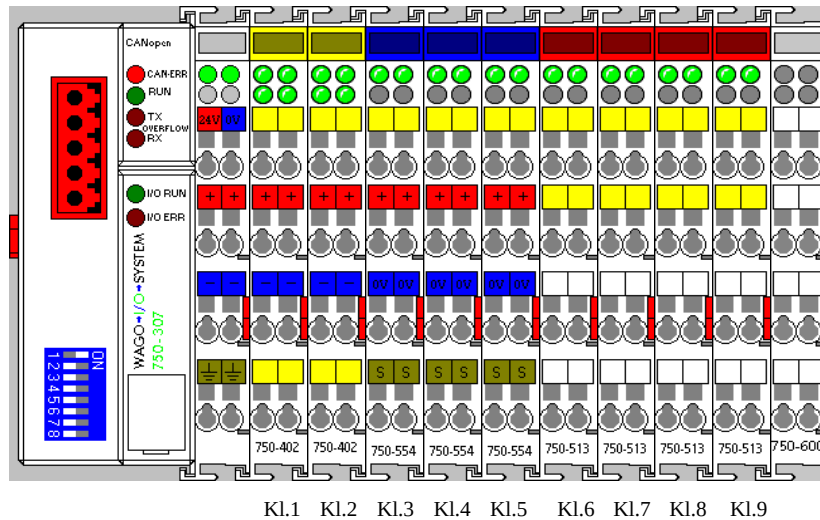
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
M	G	総バイト数						MB 長					出力	入力	A/D	

A/D	モジュールタイプがアナログかデジタルかを指定 0=アナログ、1=デジタル
入力	モジュールが入力モジュールかどうかを指定 0=入力モジュールなし、1=入力モジュール *1)
出力	モジュールが出力モジュールかどうかを指定 0=出力モジュールなし、1=出力モジュール *1)
チャンネル数	モジュールのチャンネル数を指定
ビット数/バイト数	プロセスイメージにマッピングされるチャンネル当りのバイト数（アナログモジュール）またはビット数（デジタルモジュール）を規定する
予約	予約
M	モジュールを配置するかどうかを指定 0=モジュールを置かない、1=モジュールを置く
MB 長	メールボックスのサイズを指定
総バイト数	プロセスイメージ中の全ゲートウェイモジュールをバイト数で指定（メールボックス+プロセスデータ）
G	モジュールがゲートウェイモジュールかどうかを指定 0=ゲートウェイモジュールなし、1=ゲートウェイモジュール

*1) 入力および出力ビットは同時にセットできます（診断機能付デジタル出力モジュールは入力および出力ビットを持っています）。

例：

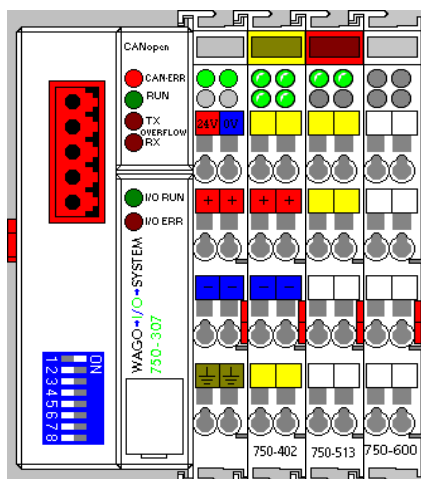
コントローラと各 I/O モジュールの構成を以下のように設定します。



全モジュール指定の場合のコンフィグレーション

S-Idx	値	意味
0	0x09	実装 9 枚、仮想 0 枚のモジュール合計
1	0x8063	デジタル入力、4 チャンネル、1 ビット/チャンネル、接続 (1 番目)
2	0x8063	デジタル入力、4 チャンネル、1 ビット/チャンネル、接続 (2 番目)
3	0x 8094	アナログ出力、2 チャンネル、2 バイト/チャンネル、接続 (3 番目)
4	0x 8094	アナログ出力、2 チャンネル、2 バイト/チャンネル、接続 (4 番目)
5	0x 8094	アナログ出力、2 チャンネル、2 バイト/チャンネル、接続 (5 番目)
6	0x 8055	デジタル出力、2 チャンネル、1 ビット/チャンネル、接続 (6 番目)
7	0x 8055	デジタル出力、2 チャンネル、1 ビット/チャンネル、接続 (7 番目)
8	0x 8055	デジタル出力、2 チャンネル、1 ビット/チャンネル、接続 (8 番目)
9	0x 8055	デジタル出力、2 チャンネル、1 ビット/チャンネル、接続 (9 番目)

次の段階として、コントローラへの実モジュール接続数を設定数より減らします。使用するモジュールを 2 番目モジュールと 8 番目モジュールのみにします。



K1.2 K1.8

2 番目と 8 番目のモジュールを接続した場合のコンフィグレーション

S-Idx	値	意味
0	0x09	実装 2 枚、仮想 7 枚のモジュール合計
1	0x0063	デジタル入力、4 チャンネル、1 ビット/チャンネル、非接続 (1 番目)
2	0x8063	デジタル入力、4 チャンネル、1 ビット/チャンネル、接続 (2 番目)
3	0x 0094	アナログ出力、2 チャンネル、2 バイト/チャンネル、非接続 (3 番目)
4	0x 0094	アナログ出力、2 チャンネル、2 バイト/チャンネル、非接続 (4 番目)
5	0x 0094	アナログ出力、2 チャンネル、2 バイト/チャンネル、非接続 (5 番目)
6	0x 0055	デジタル出力、2 チャンネル、1 ビット/チャンネル、非接続 (6 番目)
7	0x 0055	デジタル出力、2 チャンネル、1 ビット/チャンネル、非接続 (7 番目)
8	0x 8055	デジタル出力、2 チャンネル、1 ビット/チャンネル、接続 (8 番目)
9	0x 0055	デジタル出力、2 チャンネル、1 ビット/チャンネル、非接続 (9 番目)

オブジェクトディクショナリの入力値（実装モジュール）に関しては、両方のコントローラ共ビヘイビアは同じです。従って、PDO マッピング中もまたビヘイビアは同じです。例えば、8 番目出力モジュール（75-513）の 1 チャンネル目をセットできるようにするために、インデックス 0x6200、サブインデックス 1 にビット 4 をセットしなければなりません。

この手順は両方のコンフィグレーションに対して同じです。

上記と異なって空モジュールのコンフィグレーションがなかった場合、同じ出力チャンネルをセットするために、2 番目コンフィグレーションのインデックス 0x6200、サブインデックス 1 にビット 0 をセットしなければなりません。

4.3.4.3.22 オブジェクト 0x5000、入力プロセスイメージ・リード

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x5000	0	入力バイト数	Unsigned16	RO	-	入力 PA の関連バイト数
	1	入力セグメント 1	Octed_String	RO	-	1 番目入力 PA セグメント (512 バイト PA の下側 255 バイト)
	2	入力セグメント 2	Octed_String	RO	-	2 番目入力 PA セグメント (512 バイト PA の上側 255 バイト、入力データが 255 バイトを超えた場合のみ適用)

全入力データへのアクセスを 1 ブロック単位でできるようにし、SDO 経由のドメインとして全入力プロセスイメージのリードを許可します。



メモ

SDO 経由のアクセスは遅いため、時間に厳しいデータは PDO 経由でのみ送信することをお勧めします。

4.3.4.3.23 オブジェクト 0x5001、出力プロセスイメージ・ライト

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x5001	0	出力バイト数	Unsigned16	RO	none	出力 PA の関連バイト数
	1	出力セグメント 1	Octed_String	RW	none	1 番目出力 PA セグメント (512 バイト PA の下側 255 バイト)
	2	出力セグメント 2	Octed_String	RW	none	2 番目出力 PA セグメント (512 バイト PA の上側 255 バイト、モジュールコンフィグレーションに基づき出力データが 255 バイトを超える可能性がある場合のみ適用)

SDO 経由のドメインとして (ブロック単位で)、全出力プロセスイメージへのライトを許可します。



メモ

SDO 経由のアクセスは遅いため、時間に厳しいデータは PDO 経由でのみ送信することをお勧めします。

4.3.4.3.24 オブジェクト 0x5200、コントローラコンフィグレーション・オブジェクト

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x5200	0	使用サブインデックス最大値	Unsigned8	RO	-	サポートされるサブインデックスの最大値
	1	PDO 処理	Unsigned8	RW	0	受信した PDO の処理を規定する 0: ロジックデータ出力、PDO が 2 回到着次第内部バスサイクルをスタートする 1: 最終 PDO データを取得
	2	「注意レベル」点滅表示の停止	Unsigned8	RW	0	「注意レベル」の表示を開始/停止にする 0: 「注意レベル」の表示が開始する 1: 「注意レベル」の表示が停止する
	3	エマージェンシメッセージによるグローバルモジュール診断の停止	Unsigned8	RW	0	エマージェンシメッセージによるグローバルモジュール診断の送信を開始/停止する 0: 診断メッセージを送る 1: 診断メッセージを送らない

4.3.4.3.25 オブジェクト 0x5201、診断コンフィグレーション・オブジェクト

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x5200	0	使用サブインデックス最大値=モジュール数	Unsigned8	RO	-	サポートされるサブインデックスの最大値=モジュール数
	1	1 番目配置モジュールの診断ビヘイビア	Unsigned8	RW	0	バイト内の各ビットはモジュールの 1 チャンネルを表す (ビット 0 チャンネル 0、ビット 1 チャンネル 1...、ビット 7 チャンネル 7) ビット n=1: エマージェンシによる診断メッセージは送信されない (n=0~7) 全てのビットが 0 にセット (255) された場合、エマージェンシによる診断メッセージの送信はモジュールのチャンネル数に関係なく、このモジュールに対して停止される
	...					...
	64	64 番目モジュールの診断ビヘイビア	Unsigned8	RW	0	バイト内の各ビットはモジュールの 1 チャンネルを表す (ビット 0 チャンネル 0、ビット 1 チャンネル 1...、ビット 7 チャンネル 7) ビット n=1: エマージェンシによる診断メッセージは送信されない (n=0~7) 全てのビットが 0 にセット (255) された場合、エマージェンシによる診断メッセージの送信はモジュールのチャンネル数に関係なく、このモジュールに対して停止される

4.3.4.3.26 オブジェクト 0x5202、モジュールコンフィグレーション・オブジェクト



メモ

このオブジェクトは常に実際のコンフィグレーションを参照します。すなわちスペーサモジュールのコンフィグレーション使用にあたっては、実モジュールのみが対象となり、仮想モジュールは考慮しません。

Idx	S-Idx	名称	タイプ	属性	デフォルト値	意味
0x5202	0	使用サブインデックス最大値	Unsigned8	RO	-	サポートされるサブインデックスの最大値=バスコントローラを含んだ実 I/O モジュール数
	1	モジュール記述	Unsigned64	RO	-	サブインデックス 1 は常にバスコントローラの記述を参照する
	2	モジュール記述	Unsigned64	RO	-	1 番目 I/O モジュール
	3	モジュール記述	Unsigned64	RO	-	2 番目 I/O モジュール
	...	...	...	...	...	...
	65	モジュール記述	Unsigned64	RO	-	64 番目 I/O モジュール

モジュール記述の構造

フィールドバスコントローラ/アナログ I/O モジュール

ビット 63 ビット 48	ビット 47 ビット 32	ビット 31 ビット 16	ビット 15 ビット 0
シリーズ	型式番号	シリーズ拡張	型番拡張
例：フィールドバスコントローラ 750-837、I/O モジュール 750-404/000-001:			
750	837	0	0
750	404	0	1

デジタル I/O モジュール

ビット 63 ビット 48	ビット 47 ビット 32	ビット 31 ビット 16	ビット 15 ビット 0
0	0	0	デジタルモジュール記述 ビット 0...3: 0: 診断のみ 1: 入力のみ 2: 出力のみ 3: 出力+診断 4: 未使用 5: 入力+診断 6: 未使用 7: 入力+出力+診断 8: 未使用 9: 未使用 A: 未使用 B: 出力+診断 C: 未使用 D: 未使用 E: 未使用 F: 入力+出力 ビット 4...7: 未使用 ビット 8...14: 内部データ幅 (ビット単位) Bit 15: 常に 1
例：750-501:			
0	0	0	0x8202

4.3.4.4 標準デバイスプロファイル領域—DS 401

フィールドバスコントローラは標準デバイスプロファイル「汎用 I/O モジュールのデバイスプロファイル」をサポートします。

以下の表には、フィールドバスコントローラがサポートする標準プロファイル DS401 にある全てのオブジェクトを示しています。

Idx	名称	タイプ	意味	参照頁
0x6000	デジタル入力 8 ビット	Array Unsigned8	デジタル入力 I/O データを 8 ビット単位でリード	133
0x6005	グローバルインタラプトイネーブル、デジタル 8 ビット	Unsigned8	8 ビットデジタル入力データのインタラプト設定を全体的に有効/無効化	133
0x6006	インタラプトマスク、全入力変化 8 ビット	Array Unsigned8	8 ビットデジタル入力データの各々の変化により送信を実行	133
0x6007	インタラプトマスク、L-H 変化 8 ビット	Array Unsigned8	8 ビットデジタル入力データの正方向変化があれば送信を実行	133
0x6008	インタラプトマスク、H-L 変化 8 ビット	Array Unsigned8	8 ビットデジタル入力データの負方向変化があれば送信を実行	134
0x6200	デジタル出力 8 ビット	Array Unsigned8	デジタル出力 I/O モジュールのデータ	134
0x6206	エラーモード出力 8 ビット	Array Unsigned8	8 ビットデジタル出力データが定義済みエラー値に変化する/しない	135
0x6207	エラー値出力 8 ビット	Array Unsigned8	エラー値出力を 8 ビット単位でコード化し定義しておく	135
0x6401	アナログ入力 16 ビット、リード	Array Unsigned16	アナログ入力 I/O モジュールのデータ (16 ビット) をリード	135
0x6411	アナログ出力 16 ビット、ライト	Array Unsigned16	アナログ出力モジュールにデータ (16 ビット) をライト	136
0x6421	アナログ入力インタラプトトリガ選択	Array Unsigned8	各アナログ入力チャンネルに対しインタラプトを発生する条件を設定	136
0x6423	アナログ入力グローバルインタラプトイネーブル	Boolean	アナログ入力データのインタラプト設定を全体的に有効/無効化	136
0x6424	アナログ入力インタラプト、上限整数	Array Unsigned16	アナログ入力データがこれを超えたときインタラプトをトリガする上限値	137
0x6425	アナログ入力インタラプト、下限整数	Array Unsigned16	アナログ入力データがこれを超えたときインタラプトをトリガする下限値	137
0x6426	アナログ入力インタラプト、デルタ Unsigned	Array Unsigned16	アナログ入力データがこれを超えたときインタラプトをトリガする変化値	137
0x6427	アナログ入力インタラプト、負デルタ Unsigned	Array Unsigned16	アナログ入力データがこれを超えたときインタラプトをトリガする下方変化値	138
0x6428	アナログ入力インタラプト、正デルタ Unsigned	Array Unsigned16	アナログ入力データがこれを超えたときインタラプトをトリガする上方変化値	138
0x6443	アナログ出力エラーモード	Array Unsigned8	アナログ出力用の定義済みエラー値に設定するかどうかを指定	138
0x6444	アナログ出力エラー値、整数	Array Unsigned16	各アナログ出力に設定するエラー値を定義しておく	139
0x67FE	エラービヘイビア	Array Unsigned8	エラーが起きたときの状態変化	139

4.3.4.4.1 オブジェクト 0x6000、デジタル入力 8 ビット

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6000	0	デジタル入力ブロック数	Unsigned8	RO	-
	1	1 番目入力ブロック	Unsigned8	RO	-
	2	2 番目入力ブロック	Unsigned8	RO	-
					
	32	32 番目入力ブロック	Unsigned8	RO	-

このオブジェクトにはデジタル入力モジュールのプロセスデータが入ります。サブインデックス 1 には、バスコントローラから近い順に数えて最初のデジタル入力 8 チャンネル分を左から右に入れます。サブインデックス 2 には次のデジタル入力を同様に入れます。後続のモジュールも同様です。

4.3.4.4.2 オブジェクト 0x6005、グローバルインタラプトイネーブル、デジタル 8 ビット

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6005	0	グローバルインタラプトイネーブル、デジタル 8 ビット	Unsigned8	RW	1

このオブジェクトによりデジタル入力データの送信は PDO を用いてコントロールされます。値が 1 の場合送信は全体的に可能になりますが、オブジェクト 0x6006～0x6008 の内容および PDO の送信タイプにより送信されるビットが異なります。値が 0 の場合デジタル入力データは、オブジェクト 0x6006～0x6008 の内容にかかわらず送信されません。

4.3.4.4.3 オブジェクト 0x6006、インタラプトマスク、全入力変化 8 ビット

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6006	0	デジタル入力ブロック数	Unsigned8	RO	-
	1	1 番目入力ブロックのマスク	Unsigned8	RW	255
	2	2 番目入力ブロックのマスク	Unsigned8	RW	255
					
	32	32 番目入力ブロックのマスク	Unsigned8	RW	255

このオブジェクトはデジタル入力チャンネルに対し、全ての入力変化があった場合にデータを送るよう定義するのに用います。送信できるための前提条件としてはオブジェクト 0x6005 = 1 に設定します。

0=入力変化（チャンネル毎）があったときでも送信を停止します。

1=入力変化（チャンネル毎）があったとき送信を実行します。

例： サブインデックス 0=1、サブインデックス 1=65=0x41
入力変化があった場合チャンネル 1 とチャンネル 7 のみがデータを送信します。

4.3.4.4.4 オブジェクト 0x6007、インタラプトマスク、L-H 変化 8 ビット

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6007	0	デジタル入力ブロック数	Unsigned8	RO	-
	1	1 番目入力ブロックのマスク	Unsigned8	RW	0
	2	2 番目入力ブロックのマスク	Unsigned8	RW	0
					
	32	32 番目入力ブロックのマスク	Unsigned8	RW	0

このオブジェクトはデジタル入力チャンネルに対し、入力の立ち上がりが検出された（0 から 1 への変化）場合にデータを送るよう定義するのに用います。送信できるための前提条件としてはオブジェクト 0x6005=1 に設定します。このオブジェクトはオブジェクト 0x6006 と OR で動作します。

0=入力の立ち上がり（チャンネル毎）があつたときでも送信を停止します。
1=入力の立ち上がり（チャンネル毎）があつたとき送信を実行します。

例： インデックス 0x6006、サブインデックス 0=1、サブインデックス 1=65=0x41
インデックス 0x6007、サブインデックス 0=1、サブインデックス 1=33=0x21
入力変化があつた場合チャンネル 1 とチャンネル 7 は常にデータを送信します。
チャンネル 6 のみが入力の立ち上がりでデータを送信します。

4.3.4.4.5 オブジェクト 0x6008、インタラプトマスク、H-L 変化 8 ビット

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6008	0	デジタル入力ブロック数	Unsigned8	RO	-
	1	1 番目入力ブロックのマスク	Unsigned8	RW	0
	2	2 番目入力ブロックのマスク	Unsigned8	RW	0
					
	32	32 番目入力ブロックのマスク	Unsigned8	RW	0

このオブジェクトはデジタル入力チャンネルに対し、入力の立ち下がりが検出された（1 から 0 への変化）場合にデータを送るよう定義するのに用います。送信できるための前提条件としてはオブジェクト 0x6005=1 に設定します。このオブジェクトはオブジェクト 0x6006 と OR で動作します。

0=入力の立ち下がり（チャンネル毎）があつたときでも送信を停止します。
1=入力の立ち下がり（チャンネル毎）があつたとき送信を実行します。

例： インデックス 0x6006、サブインデックス 0=1、サブインデックス 1=65=0x41
インデックス 0x6008、サブインデックス 0=1、サブインデックス 1=33=0x21
入力変化があつた場合チャンネル 1 とチャンネル 7 は常にデータを送信します。
チャンネル 6 のみが入力の立ち下がりでデータを送信します。

4.3.4.4.6 オブジェクト 0x6200、デジタル出力 8 ビット

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6200	0	デジタル出力ブロック数	Unsigned8	RO	-
	1	1 番目出力ブロック	Unsigned8	RW	0
	2	2 番目出力ブロック	Unsigned8	RW	0
					
	32	32 番目出力ブロック	Unsigned8	RW	0

このオブジェクトにはデジタル出力モジュールのプロセスデータが入ります。サブインデックス 1 には、バスコントローラから近い順に数えて最初のデジタル出力 8 チャンネル分を左から右に入れます。サブインデックス 2 には次のデジタル出力を同様に入れます。後続のモジュールも同様です。

4.3.4.4.7 オブジェクト 0x6206、エラーモード出力 8 ビット

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6206	0	デジタル出力ブロック数	Unsigned8	RO	-
	1	1 番目出力ブロックのマスク	Unsigned8	RW	255
	2	2 番目出力ブロックのマスク	Unsigned8	RW	255
					
	32	32 番目出力ブロックのマスク	Unsigned8	RW	255

このオブジェクトには、エラーが起きた場合に出力を定義済みのエラーステータスに変えるかどうかを定義します（例：バスコントローラが STOPPED ステータスに移行、ノードゲーディングが失敗した、等）（オブジェクト 0x6207 参照）。エラーが修復した場合、出力は直前の状態のままとなります。すなわち出力チャンネルのエラーステータスは変化しません。

0=当該ビットの出力は変化しない（チャンネル毎）

1=当該ビットの出力は定義済みのエラーステータスに変わる（チャンネル毎）

4.3.4.4.8 オブジェクト 0x6207、エラー値出力 8 ビット

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6207	0	デジタル出力ブロック数	Unsigned8	RO	-
	1	1 番目出力ブロックのマスク	Unsigned8	RW	0
	2	2 番目出力ブロックのマスク	Unsigned8	RW	0
					
	32	32 番目出力ブロックのマスク	Unsigned8	RW	0

このオブジェクトは、エラーが起きたときに各出力にセットすべき値を定義するのに用います。この設定が有効になるためには、オブジェクト 0x6206 において対応するビットをセットしておく必要があります。

0=変化対応ビットの出力を 0 にセット（チャンネル毎）

1=変化対応ビットの出力を 1 にセット（チャンネル毎）

例： インデックス 0x6206、サブインデックス 0=1、サブインデックス 1=65=0x41
インデックス 0x6207、サブインデックス 0=1、サブインデックス 1=33=0x21
チャンネル 1 は 1 にセットされ、チャンネル 7 は 0 にセットされます。
他の全てのチャンネルはエラーが起きた場合に変化しません。

4.3.4.4.9 オブジェクト 0x6401、アナログ入力 16 ビット

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6401	0	アナログ入力チャンネル数（16 ビット単位）	Unsigned8	RO	-
	1	1 番チャンネル	Unsigned16	RO	-
					
	254	254 番チャンネル	Unsigned16	RO	-

このオブジェクトにはアナログ入力モジュールのプロセスデータが入ります。サブインデックス 1 には、バスコントローラから近い順に数えて最初のアナログ入力チャンネルを左から右に入れます。サブインデックス 2 には次のアナログ入力を同様に入れます。後続のモジュールも同様です。

4.3.4.4.10 オブジェクト 0x6411、アナログ出力 16 ビット

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6411	0	アナログ出力チャンネル数 (16 ビット単位)	Unsigned8	RO	-
	1	1 番チャンネル	Unsigned16	RW	0
					
	254	254 番チャンネル	Unsigned16	RW	0

このオブジェクトにはアナログ出力モジュールのプロセスデータが入ります。サブインデックス 1 には、バスコントローラから近い順に数えて最初のアナログ出力チャンネルを左から右に入れます。サブインデックス 2 には次のアナログ出力を同様に入れます。後続のモジュールも同様です。

4.3.4.4.11 オブジェクト 0x6421、アナログ入力インタラプトトリガ選択

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6421	0	アナログ入力チャンネル数 (16 ビット単位)	Unsigned8	RO	-
	1	1 番チャンネル用トリガ	Unsigned8	RW	7
					
	128	254 番チャンネル用トリガ	Unsigned8	RW	7

このオブジェクトはインタラプトを送信する条件を定義するのに用いられます。送信ができるためには、オブジェクト 0x6423 に 1 を入れて、送信を全体的に有効にしておく必要があります。

サブインデックス 1...128 の構造

ビット	送信条件	コンフィグレーション対象
0	入力値が閾値を超えた	0x6424
1	入力値が閾値以下になった	0x6425
2	入力値の変化が前回の送信でのデルタ値 (変化幅) を超えた	0x6426
3	入力値の減少が前回の送信でのデルタ値を越えた	0x6427
4	入力値の増加が前回の送信でのデルタ値を超えた	0x6428
5 to 7	予約	-

4.3.4.4.12 オブジェクト 0x6423、アナログ入力グローバルインタラプトイネーブル

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6423	0	グローバルインタラプトイネーブル、アナログ 16 ビット	Unsigned8	RW	0

このオブジェクトは、PDO によるアナログ入力データの送信をコントロールするのに用いられます。値が 1 の場合送信は実行されますが、送信条件はオブジェクト 0x6421 の内容によって異なり、また PDO の送信タイプによっても異なります。値が 0 の場合、オブジェクト 0x6421 の内容にかかわらず、アナログ入力データの送信は行われません。



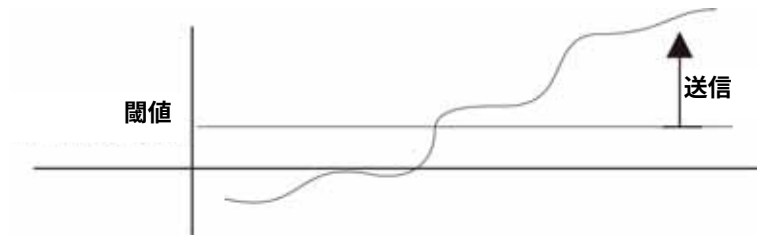
注意

デフォルトの設定では、アナログ入力データの送信は停止になっています。

4.3.4.4.13 オブジェクト 0x6424、アナログ入力インタラプト、上限整数

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6424	0	アナログ入力チャンネル数 (16 ビット)	Unsigned8	RO	-
	1	1 番チャンネル上限値	Unsigned16	RW	0
					
	128	128 番チャンネル上限値	Unsigned16	RW	0

オブジェクト 0x6423 で実行可能に設定されていた場合、このオブジェクトの内容により閾値のモニタリングが可能になります。入力値が定義した閾値以上の場合、さらにトリガ条件がセットされる（オブジェクト 0x6426 により）まで送信は行われません。



g012435e

4.3.4.4.14 オブジェクト 0x6425、アナログ入力インタラプト、下限整数

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6425	0	アナログ入力チャンネル数 (16 ビット)	Unsigned8	RO	-
	1	1 番チャンネル下限値	Unsigned16	RW	0
					
	128	128 番チャンネル下限値	Unsigned16	RW	0

オブジェクト 0x6423 で実行可能に設定されていた場合、このオブジェクトの内容により閾値のモニタリングが可能になります。入力値が定義した閾値以下の場合、さらにトリガ条件がセットされる（オブジェクト 0x6426 により）まで送信が行われます。



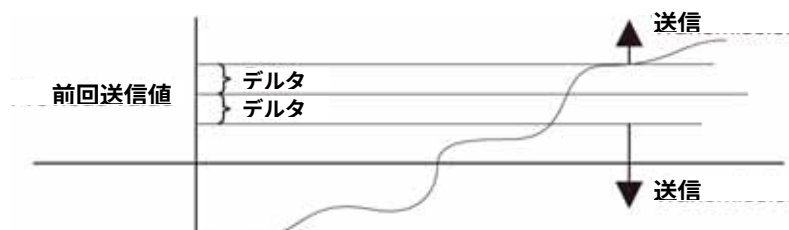
g012436e

4.3.4.4.15 オブジェクト 0x6426、アナログ入力インタラプト、デルタ Unsigned

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6426	0	アナログ入力チャンネル数 (16 ビット)	Unsigned8	RO	-
	1	1 番チャンネルデルタ値	Unsigned16	RW	0
					
	128	128 番チャンネルデルタ値	Unsigned16	RW	0

このオブジェクトの定義により、新しく送信する値は、以前送った値に対して少なくともデルタ値分以上の値か、またはデルタ値分以下の値でなければなりません。

例えばこのオブジェクトはオブジェクト 0x6424 と関連して使うことができますので、設定した閾値とデルタ機能を満たしたときのみ、送信を完了することができます。

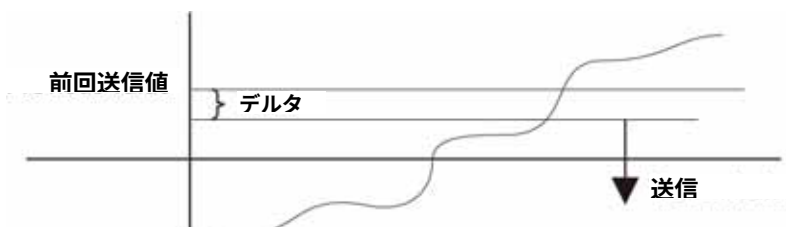


g012437e

4.3.4.4.16 オブジェクト 0x6427、アナログ入力インタラプト、負デルタ Unsigned

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6427	0	アナログ入力チャンネル数 (16 ビット)	Unsigned8	RO	-
	1	1 番チャンネルデルタ値	Unsigned16	RW	0
					
	128	128 番チャンネルデルタ値	Unsigned16	RW	0

このオブジェクトの定義により、新しく送信する値は、以前送った値に対してデルタ値分以下の値でなければなりません。

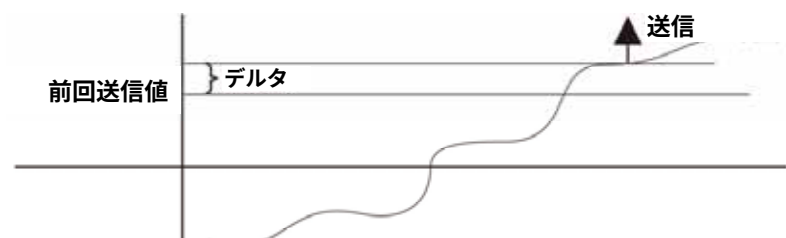


g012438e

4.3.4.4.17 オブジェクト 0x6428、アナログ入力インタラプト、正デルタ Unsigned

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6428	0	アナログ入力チャンネル数 (16 ビット)	Unsigned8	RO	-
	1	1 番チャンネルデルタ値	Unsigned16	RW	0
					
	128	128 番チャンネルデルタ値	Unsigned16	RW	0

このオブジェクトの定義により、新しく送信する値は、以前送った値に対してデルタ値分以上の値でなければなりません。



g012439e

4.3.4.4.18 オブジェクト 0x6443、アナログ出力エラーモード

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6443	0	アナログ出力チャンネル数 (16 ビット)	Unsigned8	RO	-
	1	1 番チャンネルエラーモード	Unsigned8	RW	1
					
	128	128 番チャンネルエラーモード	Unsigned8	RW	1

このオブジェクトには、エラーが起きた場合（例：バスコントローラが STOPPED ステータスに移行、ノードガーディングが失敗した、等）に出力を定義済みのエラーステータス（オブジェクト 0x6444 参照）に変えるかどうかを定義します。エラーが修復した場合、出力は直前の状態のままとなります。すなわち出力チャンネルのエラーステータスは変化しません。

オブジェクト 0x6444 が適用できない全てのアナログ出力（例：アナログ 6 バイトモジュールなど）は、エラーが起きたときは常に 0 にセットします。

0=出力は変化しません

1=出力は定義済みのエラーステータスに変わります

4.3.4.4.19 オブジェクト 0x6444、アナログ出力エラー値、整数

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x6444	0	アナログ出力チャンネル数 (16 ビット)	Unsigned8	RO	-
	1	1 番チャンネルエラー値	Unsigned16	RW	0
					
	128	128 番チャンネルエラー値	Unsigned16	RW	0

このオブジェクトにはエラーが起きた場合にとるべき値を定義します。予め、オブジェクト 0x6443 に対応するビットをセットしておく必要があります。

4.3.4.4.20 オブジェクト 0x67FE、エラービヘイビア

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0x67FE	0	サポートされるインデックスの最大値	Unsigned8	RO	1
	1	通信エラー	Unsigned8	RW	0

このオブジェクトには、通信エラー（例：ノードガーディング失敗など）が起きた場合にモジュールが移行するステータスを定義します。

通信エラー入力値の意味

通信エラー	動作
0	PRE-OPERATIONAL ステータスに移行（現在のステータスが OPERATIONAL だったときのみ）
1	ステータス変化なし
2	STOPPED ステータスに移行

4.3.4.5 予約領域

IEC61131-3 変数のデータは、750-837/750-838 プログラマブルバスコントローラで使用されるオブジェクトディクショナリの予約領域 (インデックス 0xA000-0xFFFF) に入力します。全体としては、インデックス 0xA000~0xA8FF が IEC61131-3 変数データの割り当てに用いられます。



メモ

IEC61131-3 入力変数は CAN バスの立場から定義していますが、PFC の方からは出力変数とみなされます。従って、IEC61131-3 出力変数は PFC にとっては入力変数になります。

以下の表には、IEC61131-3 変数のために 750-837/750-838 フィールドバスコントローラがサポートするオブジェクトの全てが入っています。

Idx	名称	タイプ	意味	参照頁
0xA000	Integer 8 入力変数	Array Integer 8	8 ビット整数入力変数データ	141
0xA040	Unsigned 8 入力変数	Array Unsigned 8	8 ビット Unsigned 入力変数データ	141
0xA080	Boolean 入力変数	Boolean	Boolean 入力変数データ	141
0xA0C0	Integer 16 入力変数	Array Integer 16	16 ビット整数入力変数データ	141
0xA100	Unsigned 16 入力変数	Array Unsigned 16	16 ビット Unsigned 入力変数データ	142 (以下 同様)
0xA140	Integer 24 入力変数	Array Integer 24	24 ビット整数入力変数データ	
0xA180	Unsigned 24 入力変数	Array Unsigned 24	24 ビット Unsigned 入力変数データ	
0xA1C0	Integer 32 入力変数	Array Integer 32	32 ビット整数入力変数データ	
0xA200	Unsigned 32 入力変数	Array Unsigned 32	32 ビット Unsigned 入力変数データ	
0xA240	Floating Point 32 入力変数	Array Floating Point 32	32 ビット Floating Point 入力変数データ	
0xA280	Unsigned 40 入力変数	Array Unsigned 40	40 ビット Unsigned 入力変数データ	
0xA2C0	Integer 40 入力変数	Array Integer 40	40 ビット整数入力変数データ	
0xA300	Unsigned 48 入力変数	Array Unsigned 48	48 ビット Unsigned 入力変数データ	
0xA340	Integer 48 入力変数	Array Integer 48	48 ビット整数入力変数データ	
0xA380	Unsigned 56 入力変数	Array Unsigned 56	56 ビット Unsigned 入力変数データ	
0xA3C0	Integer 56 入力変数	Array Integer 56	56 ビット整数入力変数データ	
0xA400	Integer 64 入力変数	Array Integer 64	64 ビット整数入力変数データ	
0xA440	Unsigned 64 入力変数	Array Unsigned 64	64 ビット Unsigned 入力変数データ	
0xA480	Integer 8 出力変数	Array Integer 8	8 ビット整数出力変数データ	
0xA4C0	Unsigned 8 出力変数	Array Unsigned 8	8 ビット Unsigned 出力変数データ	
0xA500	Boolean 出力変数	Boolean	Boolean 出力変数データ	
0xA540	Integer 16 出力変数	Array Integer 16	16 ビット整数出力変数データ	
0xA580	Unsigned 16 出力変数	Array Unsigned 16	16 ビット Unsigned 出力変数データ	
0xA5C0	Integer 24 出力変数	Array Integer 24	24 ビット整数出力変数データ	
0xA600	Unsigned 24 出力変数	Array Unsigned 24	24 ビット Unsigned 出力変数データ	
0xA640	Integer 32 出力変数	Array Integer 32	32 ビット整数出力変数データ	
0xA680	Unsigned 32 出力変数	Array Unsigned 32	32 ビット Unsigned 出力変数データ	
0xA6C0	Floating Point 32 出力変数	Array Floating Point 32	32 ビット Floating Point 出力変数データ	
0xA700	Unsigned 40 出力変数	Array Unsigned 40	40 ビット Unsigned 出力変数データ	
0xA740	Integer 40 出力変数	Array Integer 40	40 ビット整数出力変数データ	
0xA780	Unsigned 48 出力変数	Array Unsigned 48	48 ビット Unsigned 出力変数データ	
0xA7C0	Integer 48 出力変数	Array Integer 48	48 ビット整数出力変数データ	
0xA800	Unsigned 56 出力変数	Array Unsigned 56	56 ビット Unsigned 出力変数データ	
0xA840	Integer 56 出力変数	Array Integer 56	56 ビット整数出力変数データ	
0xA880	Integer 64 出力変数	Array Integer 64	64 ビット整数出力変数データ	
0xA8C0	Unsigned 64 出力変数	Array Unsigned 64	64 ビット Unsigned 出力変数データ	

4.3.4.5.1 オブジェクト 0xA000、Integer8 IEC61131-3 入力変数

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0xA000	0	入力データブロック数	Unsigned8	RO	-
	1	1 番目入力データブロック	Integer8	RO	-
	2	2 番目入力データブロック	Integer8	RO	-
....					
0xA002	2	512 番目入力データブロック	Integer8	RO	-

このオブジェクトには Integer8 タイプの入力変数のプロセスデータが入ります。サブインデックス 1 には 1 番目の 8 ビット入力データが入ります。サブインデックス 2 以降には後続の入力データが入ります。

4.3.4.5.2 オブジェクト 0xA040、Unsigned8 IEC61131-3 入力変数

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0xA040	0	入力データブロック数	Unsigned8	RO	-
	1	1 番目入力データブロック	Unsigned8	RO	-
	2	2 番目入力データブロック	Unsigned8	RO	-
....					
0xA042	2	512 番目入力データブロック	Unsigned8	RO	-

このオブジェクトには Unsigned8 タイプの入力変数のプロセスデータが入ります。サブインデックス 1 には 1 番目の 8 ビット入力データが入ります。サブインデックス 2 以降には後続の入力データが入ります。

4.3.4.5.3 オブジェクト 0xA080、Boolean IEC61131-3 入力変数

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0xA080	0	入力データブロック数	Unsigned8	RO	-
	1	1 番目入力データブロック	Boolean	RO	-
	2	2 番目入力データブロック	Boolean	RO	-
....					
0xA082	2	512 番目入力データブロック	Boolean	RO	-

このオブジェクトには Boolean タイプの入力変数のプロセスデータが入ります。サブインデックス 1 には 1 番目の 8 ビット入力データが入ります。サブインデックス 2 以降には後続の入力データが入ります。

4.3.4.5.4 オブジェクト 0xA0C0、Integer16 IEC61131-3 入力変数

Idx	S-Idx	名称	タイプ	属性	デフォルト値
0xA0C0	0	入力データブロック数	Unsigned8	RO	-
	1	1 番目入力データブロック	Integer16	RO	-
	2	2 番目入力データブロック	Integer16	RO	-
....					
0xA0C1	1	256 番目入力データブロック	Integer16	RO	-

このオブジェクトには Integer16 タイプの入力変数のプロセスデータが入ります。サブインデックス 1 には 1 番目の 16 ビット入力データが入ります。サブインデックス 2 以降には後続の入力データが入ります。

4.3.4.5.5 オブジェクト 0xA101~0xA8C0、IEC61131-3 入力/出力変数

同様に、以下のオブジェクト全ての構造は既に記述したオブジェクトのものと一致しています。

各インデックスには最大 256 のサブインデックス (S-Idx 0~255) があります。

サブインデックス 0 にはデータ入力の数が入り、以降のサブインデックスにはデータがブロック単位で入ります。ブロックのサイズは、モジュールのデータタイプ毎のデータ幅によって異なります。

S-Idx	内 容
0	データブロック数
1	1 番目データブロック、相当するデータタイプのデータ幅が入る
2	2 番目データブロック、相当するデータタイプのデータ幅が入る
....	

インデックスおよびサブインデックスの最大値は、バスコントローラのメモリ容量が 512 バイトであることと個々のデータタイプ毎のデータ幅によって決まります。



詳細情報

変数データのインデックス最大値を調べる場合、750-837/750838 プログラマブルバスコントローラをカバーしている第 3.1.6.4.1.4 節「最大インデックス」を参照してください。

4.3.5 PDO 送信

PDO のデータ送信は OPERATIONAL ステータスでのみ可能です。

OPERATIONAL ステータスに移行したとき、全ての TxPDO は送信タイプ 254 および 255 で 1 度だけ送信されます。



メモ

特別送信タイプ 254 および 255（インデックス 0x1800—0x181F、サブインデックス 2）：

アナログの変化はデバイスプロファイル DS401、オブジェクト 0x6423（アナログ入力グローバルインタラプトイネーブル）で定義したデフォルト値（=FALSE）のため送信されません。このようにすると、CAN メッセージによる CAN バスのオーバーフローを防ぐことができます。オブジェクト 0x6423=TRUE をセットしたときオーバーフローを防ぐためには、対応した長い Inhibit タイムを選択することができます。さらに閾値モニタリング用（オブジェクト 0x6421、0x6424、0x6425）およびデルタ機能用（オブジェクト 0x6426、0x6427、0x6428）のオブジェクトをコンフィグレーションすることによりメッセージ量を減らすことができます。

4.3.5.1 マッピング

PDO のマッピングによって、PDO を使用して送信するデータを定義することができます。

保存されたユーザ固有のコンフィグレーションが用いられない場合や他の設定が一切されていない場合、オブジェクトディクショナリはデバイスプロファイル DS401 に基づいたデフォルトコンフィグレーション（第 4.3.4.1 節「初期化」を参照）で割り当てられます。

バスコントローラが PRE-OPERATIONAL ステータスであるならば、マッピングは代わりに SDO を介し、アプリケーション固有の方法で変更することができます。



詳細情報

アプリケーション固有のマッピングコンフィグレーションを生成する方法の例に関しては第 3.1.8 節「CANopen フィールドバスノードの立ち上げ」を参照してください。

4.3.5.2 送信 PDO1

以下の図では、デジタル入力の PDO 送信に対して、関連したオブジェクトとそれらの関係の概観を示しています。

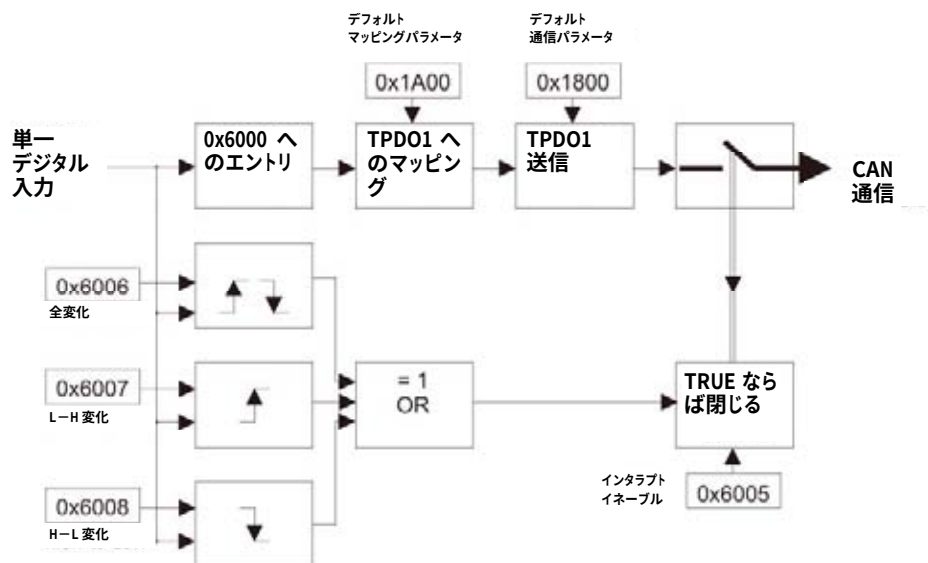


図 4-23 : デジタル入力データの PDO 送信

g012456x

インデックス	オブジェクト名	意味	参照頁
0x1800	送信 PDO 通信パラメータ	送信 PDO 用の通信パラメータ	118
0x1A00	送信 PDO マッピングパラメータ	送信 PDO 用のマッピングパラメータ	119
0x6000	デジタル入力 8 ビット	デジタル入力のデータ	133
0x6005	グローバルインタラプトイネーブル、デジタル 8 ビット	8 ビットデジタル入力データの送信を全体的に実行	133
0x6006	インタラプトマスク、入力変化 8 ビット	8 ビットデジタル入力データの全ての変化に対し送信を実行	133
0x6007	インタラプトマスク、L-H 変化 8 ビット	8 ビットデジタル入力データの L-H の立ち上がりに対し送信を実行	133
0x6008	インタラプトマスク、H-L 変化 8 ビット	8 ビットデジタル入力データの H-L の立下りに対し送信を実行	134

4.3.5.3 受信 PDO1

以下の図では、デジタル出力の PDO 送信に対して、関連したオブジェクトとそれらの関係の概観を示しています。

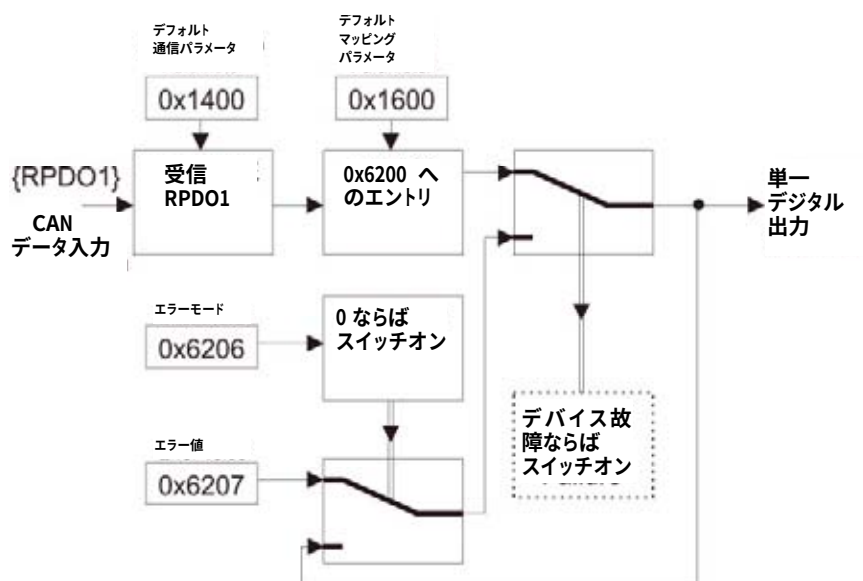


図 4-24 : デジタル出力データの PDO 送信

g012457x

インデックス	オブジェクト名	意味	参照頁
0x1400	受信 PDO 通信パラメータ	受信 PDO 用通信パラメータ	118
0x1600	受信 PDO マッピングパラメータ	受信 PDO 用マッピングパラメータ	118
0x6200	デジタル出力 8 ビット	デジタル出力のデータ	134
0x6206	エラーモード出力 8 ビット	8 ビットデジタル出力データの既定エラー値の実行	134
0x6207	エラー値出力 8 ビット	8 ビットデジタル出力データの既定エラー値	135

4.3.5.4 送信 PDO2

以下の図では、アナログ入力 の PDO 送信に対して、関連したオブジェクトとそれらの関係の概観を示しています。

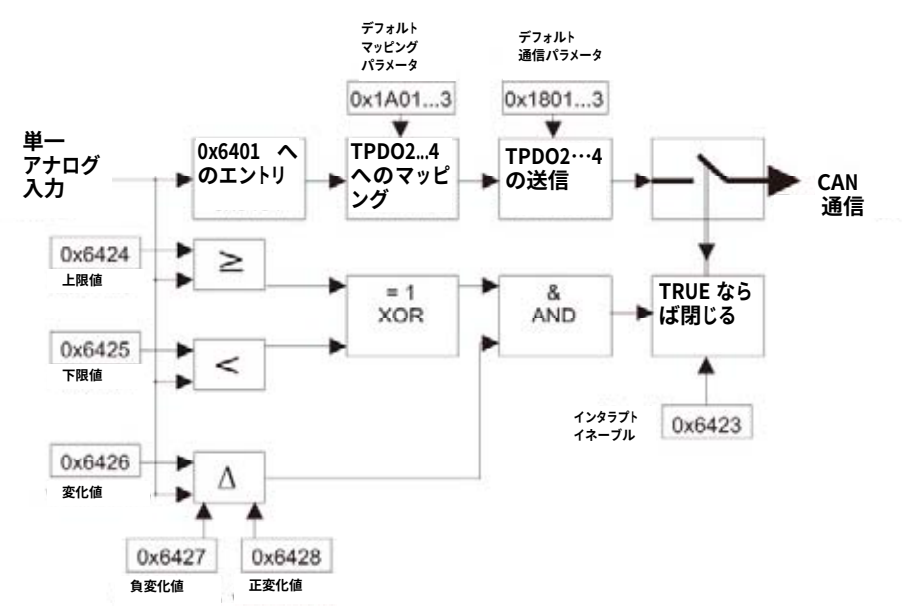


図 4-25 : アナログ入力データの PDO 送信 g012458x

インデックス	オブジェクト名	意味	参照頁
0x1801...3	送信 PDO 通信パラメータ	送信 PDO 用通信パラメータ	118
0x1A01...3	送信 PDO マッピングパラメータ	送信 PDO 用マッピングパラメータ	119
0x6401	アナログ入力 16 ビット	16 ビットアナログ入力データ	136
0x6423	アナログ入力グローバルインタラプトイネーブル	アナログ入力データの送信を全体的に実行	137
0x6424	アナログ入力インタラプト、上限整数	アナログ入力データが上限値を超えた場合これを送信	137
0x6425	アナログ入力インタラプト、下限整数	アナログ入力データが下限値を超えた場合これを送信	137
0x6426	アナログ入力インタラプト、デルタ Unsigned	アナログ入力データが前回送信値よりデルタ分正または負方向に超えた場合	137
0x6427	アナログ入力インタラプト、負デルタ Unsigned	アナログ入力データが前回送信値よりデルタ分負方向に超えた場合	138
0x6428	アナログ入力インタラプト、正デルタ Unsigned	アナログ入力データが前回送信値よりデルタ分正方向に超えた場合	138

4.3.5.5 受信 PDO2

以下の図では、アナログ出力の PDO 送信に対して、関連したオブジェクトとそれらの関係の概観を示しています。

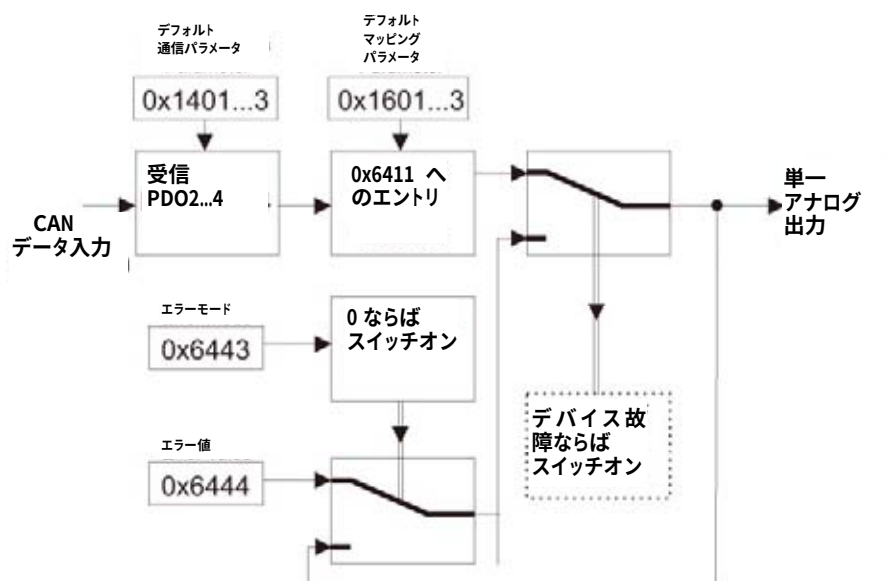


図 4-26 : アナログ出力データの PDO 送信

g012459x

インデックス	オブジェクト名	意味	参照頁
0x1401...3	受信 PDO 通信パラメータ	受信 PDO 用通信パラメータ	118
0x1601...3	受信 PDO マッピングパラメータ	受信 PDO 用マッピングパラメータ	118
0x6411	アナログ出力 16 ビット	16 ビットアナログ出力のデータ s	135
0x6443	アナログ出力エラーモード	アナログ出力データの定義済みエラー値の指示	138
0x6444	アナログ出力エラー値、整数	アナログ出力データの定義済みエラー値	139

4.3.6 SYNC 監視

通信周期の値が 0 でない場合、バスコントローラが OPERATIONAL ステータスであるならば、最初の SYNC メッセージが到着したときに監視（モニタリング）が行われます。

SYNCメッセージのエラー

監視時間（通信周期）内に SYNC メッセージを受信しなかった場合、点滅コードでこれを知らせます。このときステータスは変化しません。これに加えてエマージェンシメッセージ（エラーコード：0x8100、エラーレジスタ：0x81、追加コード：00 04 00 00 00）が送られます。SYNC メッセージのエラーは、マスタがステータス変更を要求した場合でも表示されます。

OPERATIONAL ステータスで SYNC メッセージを繰り返し受信した後にのみ、LED は正常動作状態の表示に戻ります。また更に、エマージェンシメッセージ（エラーコード：0x0000、エラーレジスタ：0x81、追加コード：00 04 00 00 00）を SYNC 監視機能が再開したことを示すために送信されます。

4.3.7 ノードガーディング

ノードガーディングは、NMT マスタが NMT スレーブ（バスコントローラ）を一定間隔でポーリングすることによって実行されます。このポーリングはリモート送信要求メッセージによって行われます。この一定間隔をガードタイム（インデックス 0x100C）と呼びます。

バスコントローラにとってノードガーディングは、マスタから最初のリモート送信要求メッセージ（RTR）のノードガーディング用 COB ID（0x700+モジュール ID）を受信したときに開始します。もしバスコントローラが相当するメッセージを受信しなかった場合、バスコントローラによるノードガーディングの監視は行われません。

デフォルト設定ではノードガーディングは動作停止になっています。つまり対応インデックス（0x100C=ガードタイム、0x100D=ライフタイムファクタ）に 0 が入っているからです。

NMT マスタがバスコントローラをポーリングしたとき、バスコントローラは応答メッセージを返します。応答メッセージには、バスコントローラの内部ステータスとトグルビットが入っています。

ガードタイムを設定せずに RTR 要求が到着した場合、ノードガーディングの監視は行われません。しかし、バスコントローラは内部ステータスを入れて応答します。

バスコントローラのステータスは以下のように各コードに対応しています。

ステータス	値
PRE-OPERATIONAL	127
OPERATIONAL	5
STOPPED	4

ライフタイム（ノード寿命）はガードタイム（インデックス 0x100C）とライフタイムファクタ（インデックス 0x100D）の積で与えられます。

ノードガーディングのエラー

ライフタイム内にノードガーディング・メッセージを受信しなかった場合、点滅コードでこれが表示されます。これに加え、エマージェンシメッセージ（エラーコード：0x8130、エラーレジスタ：0x11、追加コード 0x00 04 00 00 00）が送信され、出力はオブジェクト 0x6206、0x6207、0x6443 および 0x6444 の内容に基づいて起動され、バスコントローラはオブジェクト 0x67FE の内容に従って既定のステータスに移行します。

ノードガーディング・プロトコルが記録され次第、更にエマージェンシメッセージ（エラーコード：0x0000、エラーレジスタ：0x11、追加コード：00 04 00 00 00）をノードガーディングが再起動されたことを示すために送信されます。これにより出力とバスコントローラのステータスは以前の状態を維持します。

ノードガーディング・プロトコルまたはハートビート・プロトコルだけを使用することは可能です。ハートビート・プロデューサタイムが組み込まれた場合ハートビート・プロトコルが常に用いられます。

4.3.8 ハートビート監視

このプロトコルにより、RTR フレームを使用せずにモジュールの監視が可能になります。

ハートビート発生局は周期的にメッセージ（オブジェクト 0x1017 で定義した時間間隔）を発生し、その中でモジュールステータスを送信します。送信はオブジェクト 0x1017 を設定した後ただちに開始します。メッセージは 1 台以上のハートビートコンシューマ（オブジェクト 0x1016）によって診断することができます。最大 5 モジュールを監視できます。ハートビート・メッセージ（監視されるモジュール毎に分かれる）が最初に到着したときに監視を開始します。

ハートビートのエラー

設定時間（オブジェクト 0x1016）内に対応するハートビート・メッセージが到着しない場合、点滅コードによってこれが通知されます。これに加え、エマージェンシメッセージ（エラーコード：0x8130、エラーレジスタ：0x11、追加コード：0x00 05 KK 00 00・・・KK=EMCY をトリガしたノード番号）が送信されます。出力はオブジェクト 0x6206、0x6207、0x6443 および 0x6444 の内容に基づいて起動され、バスコントローラは 0x67FE の内容に従って定義済みのステータスに移行します。

ハートビート・プロトコルが記録され次第、更にエマージェンシメッセージ（エラーコード：0x0000、エラーレジスタ：0x11、追加コード：0x00 05 KK 00 00）をハートビートが再起動されたことを示すために送信されます。これにより出力とバスコントローラのステータスは以前の状態を維持します。複数のモジュールが監視された場合、以前のハートビートが再開した後のみ、ハートビートのエラーを知らせる点滅コードがストップします。

使用可能なプロトコルはノードガーディングまたはハートビート・プロトコルのどちらかです。ハートビート・プロトコルは、ハートビート・プロデューサタイムが設定されたときはいつでも使用されます。

4.3.9 エラーメッセージ (エマージェンシ)

エマージェンシメッセージは重大なエラー状態がデバイスに発生し、対処しなければならないときや、他のデバイスに重要な情報を伝える必要がある場合は常に送信されます。

エマージェンシ・オブジェクトの構造や意味は次ページ「EMCY コード」表に説明されており、下位バイトー上位バイトの順でバスメッセージにコード化されます。

エマージェンシ・オブジェクトはエラーが修復した後もまた送信されます（エラーコード＝0x0000、エラーレジスタおよび追加コードの内容は「EMCY コード」表に記述されています）。

ロードされた設定がデフォルト値の場合、電源投入後エマージェンシ・オブジェクトが送信されます。これは次の2つの理由により生じます。

- 設定がまだセーブされていない（インデックス 0x1010）
- セーブされた設定がバスコントローラで廃棄された。理由はモジュールが接続または抜かれたため。

EMCY コード表

バイト	0	1	2	3	7	
名称	エラーコード	エラーレジスタ	追加コード	意 味		
	0x0000*	0x00	00 00 00 00 00	「定義済みエラー領域」インデックス 0x1003 サブインデックス 0 が"0"にセットされる、または全てのエラーがクリアされる。		
	0x5000*	0x81	00 01 00 00 00	電源投入後またはノード//通信のリセット後ハードウェア構成が変更された。 バスコントローラは初期化される。理由：保存したコンフィグレーションがない、または内部コンフィグレーションが実装コンフィグレーションと一致せず。		
	0x5000*	0x81	00 02 00 00 00	フラッシュエラー フラッシュにコンフィグレーションをセーブしたときエラーが発生。		
	0x5000*	0x81	00 03 PP LL SS	プログラムしたコンフィグレーションが実装品と一致せず。 PP：エラーが起きた実際のモジュールスロット位置 LL：エラーが起きたロジック上モジュールスロット位置 SS：エラー原因		
	0x5000*	0x81	00 09 00 00 00	エマーゲンシ・メッセージのキューオーバーフロー (エマーゲンシ用 Inhibit タイムが用いられたときのみ起こりうる)		
	0x5000*	0x81	00 0A 01 00 00	ゲートウェイモジュールが最大数を越えた、またはゲートウェイモジュールによるプロセスイメージが最大サイズを越えた。		
	0x5000*	0x81	00 0A 02 00 00	メールボックスが最大サイズを越えた。		
	0x8100*	0x81	00 04 00 00 00	2 つの Sync オブジェクト間の時間間隔が通信周期より長い		
	0x8110*	0x11	00 01 00 00 00	内部受信バッファオーバーフロー オブジェクト 0x67FE で定義したステータス変化。出力はエラーモード/エラー値オブジェクトで定義した通り切り換わる。		
	0x8110*	0x11	00 02 00 00 00	内部送信バッファオーバーフロー オブジェクト 0x67FE で定義したステータス変化。出力はエラーモード/エラー値オブジェクトで定義した通り切り換わる。		
	0x8120*	0x11	00 03 00 00 00	CAN コントローラがエラーパッシブモードである		
	0x8130*	0x11	00 04 00 00 00	2 つのノードガーディング・テレグラム間の時間がガードタイム×ライフタイムファクタより大きい		
	0x8130*	0x11	00 05 KK 00 00	2 つのハートビート・テレグラム間の時間が設定値より大きい KK：時間オーバーフローを通知したノード		
	0x8210*	0x81	00 05 SS II NN	PDO が通信プロファイルで設定した値より小さいバイト数で送信された PDO データは捨てられる。出力は変化せず。 SS：セット箇所値 - 設定値 (インデックス 0x1600 サブインデックス 0) II：実際値 - 送信バイト数 NN：PDO 数 (1...32)		
	0x8220*	0x81	00 08 SS II NN	PDO が通信プロファイルで設定した値より大きいバイト数で送信された 最初の n データのみが用いられる (n=オブジェクトディクショナリで設定した全長) SS：セット箇所値 - 設定値 (全ての有効に設定したオブジェクトの総バイト数) II：実際値 - 送信バイト数 NN：PDO 数 (1...32)		
	0xFF00*	0x81	00 06 PP 00 00	内部バスエラー、STOPPED ステータスに移行 PP：モジュール位置		
	0xFF00*	0x81	DD 07 PP SK NN	診断メッセージ - DD：診断バイト - PP：モジュール位置 - SK：エラーステータスおよびチャンネル番号 - NN：現在のモジュールエラー数		

* エラーコード：バイト 0=下位バイト、バイト 1=上位バイト

例：エラーコード 0x8220： バイト 0=0x20、バイト 1=0x82

4.3.9.1 I/O モジュールの診断メッセージ

診断機能付きのモジュールでエラーが起きた場合、診断ステータスはエマージェンシ・メッセージによって送信されます。

診断メッセージ用追加コードの構造：

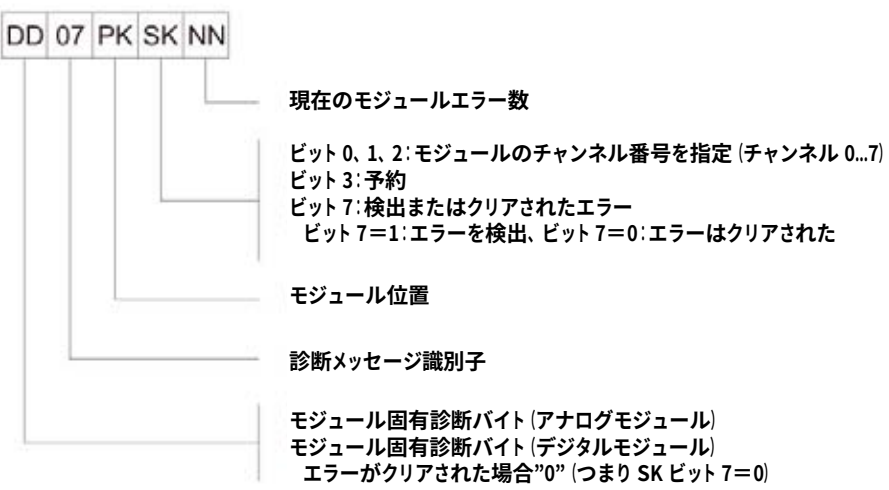


図 4-27：追加コードの構造 g012423e

例：

- 2 チャンネルアナログ入力モジュール 750-465、14 番位置に接続、チャンネル 0 の電流は 20mA 以上

エマージェンシ・テレグラム

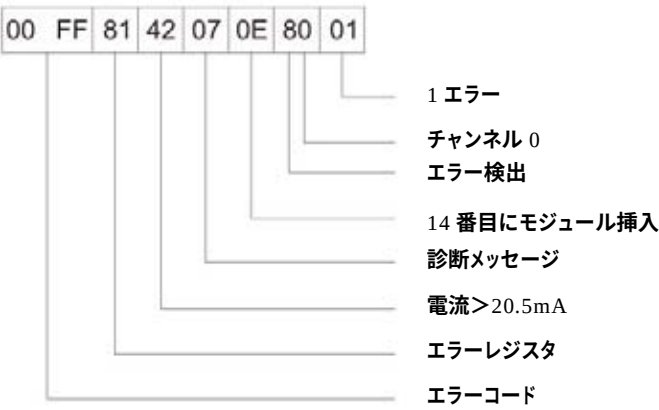


図 4-28：エマージェンシ・テレグラム 1 g012424e

- 最初のエラーに加えて、2 チャンネルデジタル出力モジュール 750-506 に別のエラーが起きます。チャンネル 1 で断線、モジュールは 17 番目に接続。

エマージェンシ・テレグラム



図 4-29 : エマージェンシ・テレグラム 2 g012425e

- 発生したエラー（デジタルモジュール 750-506 で断線）は修復されます。

エマージェンシ・テレグラム

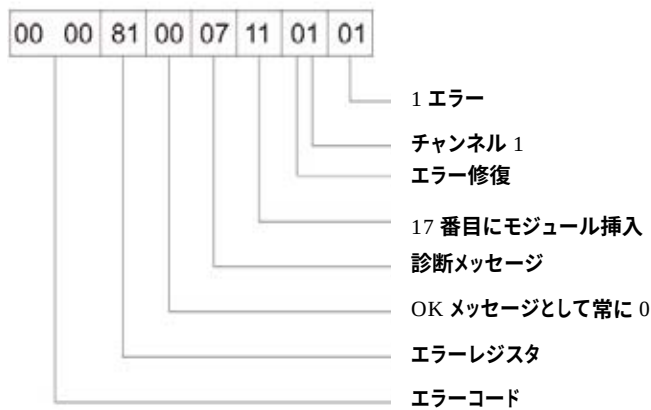


図 4-30 : エマージェンシ・テレグラム 3 g012426x



メモ
診断付のデジタルモジュールに対し、診断データはプロセスイメージに追加でマッピングされます。上記の例で用いた 750-506 は出力プロセスイメージで 4 ビット（1 ビット＝出力値、1 ビット＝未使用/チャンネル）をまた入力プロセスイメージで 4 ビット（2 ビット＝診断/チャンネル・・・断線、短絡）を使用しています。

5 I/O モジュール

5.1 概要

次節以降に掲げたバスモジュールの全ては、ワゴ I/O システム 750 のモジュラーアプリケーションに使用可能です。

I/O モジュールの詳細およびモジュールの種類に関しては「I/O モジュール取扱説明書」をご覧ください。

CD-ROM または以下の WEB サイトから入手可能です。

http://www.wago.co.jp/io/11_download_Japanese.htm



詳細情報

ワゴ I/O システムに関する最新の情報は以下のサイトから入手可能です。

http://www.wago.com/cps/rde/xchg/wago/style.xsl/gle_index.html

5.1.1 デジタル入力モジュール

DI DC 5V	
750-414	4 チャンネル、DC5V、0.2ms、2～3 線接続、正方向スイッチ
DI DC 5(12)V	
753-434	8 チャンネル、DC5(12)V、0.2ms、1 線接続、正方向スイッチ
DI DC 24V	
750-400, 753-400	2 チャンネル、DC24V、3.0ms、2～4 線接続、正方向スイッチ
750-401, 753-401	2 チャンネル、DC24V、0.2ms、2～4 線接続、正方向スイッチ
750-410, 753-410	2 チャンネル、DC24V、3.0ms、2～4 線接続、正方向スイッチ
750-411, 753-411	2 チャンネル、DC24V、0.2ms、2～4 線接続、正方向スイッチ
750-418, 753-418	2 チャンネル、DC24V、3.0ms、2～3 線接続、正方向スイッチ、診断付
750-421, 753-421	2 チャンネル、DC24V、3.0ms、2～3 線接続、正方向スイッチ、診断付
750-402, 753-402	4 チャンネル、DC24V、3.0ms、2～3 線接続、正方向スイッチ
750-403, 753-403	4 チャンネル、DC24V、0.2ms、2～3 線接続、正方向スイッチ
750-432, 753-432	4 チャンネル、DC24V、3.0ms、2 線接続、正方向スイッチ
750-433, 753-433	4 チャンネル、DC24V、0.2ms、2 線接続、正方向スイッチ
750-422, 753-422	4 チャンネル、DC24V、2～3 線接続、正方向スイッチ、10ms パルス拡張
750-408, 753-408	4 チャンネル、DC24V、3.0ms、2～3 線接続、負方向スイッチ
750-409, 753-409	4 チャンネル、DC24V、0.2ms、2～3 線接続、負方向スイッチ
750-430, 753-430	8 チャンネル、DC24V、3.0ms、1 線接続、正方向スイッチ
750-431, 753-431	8 チャンネル、DC24V、0.2ms、1 線接続、正方向スイッチ
750-436	8 チャンネル、DC24V、3.0ms、1 線接続、負方向スイッチ
750-437	8 チャンネル、DC24V、0.2ms、1 線接続、負方向スイッチ
DI AC/DC 24V	
750-415, 753-415	4 チャンネル、AC/DC24V、2 線接続
750-423, 753-423	4 チャンネル、AC/DC24V、2～3 線接続、電源ジャンプ接点付
DI AC/DC 42V	
750-428, 753-428	4 チャンネル、AC/DC42V、2 線接続
DI DC 48V	
750-412, 753-412	2 チャンネル、DC48V、3.0ms、2～4 線接続、正方向スイッチ
DI DC 110V	
750-427, 753-427	2 チャンネル、DC110V、正方向または負方向スイッチ設定可能

DI AC 120V	
750-406, 753-406	2 チャンネル、AC120V、2～4 線接続、正方向スイッチ
DI AC 120(230)V	
753-440	4 チャンネル、AC120(230)V、2 線接続、正方向スイッチ
DI AC 230V	
750-405, 753-405	2 チャンネル、AC230V、2～4 線接続、正方向スイッチ
DI NAMUR	
750-435	1 チャンネル、NAMUR EExi、DIN EN 50227 準拠近接スイッチ
750-425, 753-425	2 チャンネル、NAMUR、DIN EN 50227 準拠近接スイッチ
750-438	2 チャンネル、NAMUR EExi、DIN EN 50227 準拠近接スイッチ
DI 侵入検知	
750-424, 753-424	2 チャンネル、DC 24V、侵入検知

5.1.2 デジタル出力モジュール

DO DC 5V	
750-519	4 チャンネル、DC5V、20mA、短絡保護付、正方向スイッチ
DO DC 12(14)V	
753-534	8 チャンネル、DC12(14)V、1A、短絡保護付、正方向スイッチ
DO DC 24V	
750-501, 753-501	2 チャンネル、DC24V、0.5A、短絡保護付、正方向スイッチ
750-502, 753-502	2 チャンネル、DC24V、2.0A、短絡保護付、正方向スイッチ
750-506, 753-506	2 チャンネル、DC24V、0.5A、短絡保護付、正方向スイッチ、診断付
750-508	2 チャンネル、DC24V、2.0A、短絡保護付、正方向スイッチ、診断付
750-535	2 チャンネル、DC24V、EExi、短絡保護付、PNP 正方向スイッチ
750-504, 753-504	4 チャンネル、DC24V、0.5A、短絡保護付、正方向スイッチ
750-531, 753-531	4 チャンネル、DC24V、0.5A、2 線接続、短絡保護付、正方向スイッチ
750-516, 753-516	4 チャンネル、DC24V、0.5A、短絡保護付、負方向スイッチ
750-530, 753-530	8 チャンネル、DC24V、0.5A、短絡保護付、正方向スイッチ
750-537	8 チャンネル、DC24V、0.5A、短絡保護付、正方向スイッチ、診断付
750-536	8 チャンネル、DC24V、0.5A、短絡保護付、負方向スイッチ
DO AC 120(230)V	
753-540	4 チャンネル、AC120(230)V、0.25A、短絡保護付、正方向スイッチ
DO AC/DC 230V	
750-509、753-509	2 チャンネル、ソリッドステートリレー、AC/DC230V、300mA
750-522	2 チャンネル、ソリッドステートリレー、AC/DC230V、500mA、3A (<30 秒)
DO リレー	
750-523	1 チャンネル、AC230V、AC16A、絶縁出力、1a 接点、双安定、手動操作
750-514、753-514	2 チャンネル、AC125V、AC0.5A、DC30V、DC1A、絶縁出力、2c 接点
750-517、753-517	2 チャンネル、AC230V、1 A、絶縁出力、2c 接点
750-512、753-512	2 チャンネル、AC230V、DC30V、AC/DC2A、ノンフローティング、2a 接点
750-513、753-513	2 チャンネル、AC230V、DC30V、AC/DC2A、絶縁出力、2a 接点

5.1.3 アナログ入力モジュール

AI 0-20mA	
750-452、753-452	2 チャンネル、0-20mA、差動入力
750-465、753-465	2 チャンネル、0-20mA、シングルエンド
750-472、753-472	2 チャンネル、0-20mA、16 ビット、シングルエンド
750-480	2 チャンネル、0-20mA、差動入力
750-453、753-453	4 チャンネル、0-20mA、シングルエンド
AI 4-20mA	
750-454、753-454	2 チャンネル、4-20mA、差動入力
750-474、753-474	2 チャンネル、4-20mA、16 ビット、シングルエンド
750-466、753-466	2 チャンネル、4-20mA、シングルエンド
750-485	2 チャンネル、4-20mA、EExi、シングルエンド
750-492、753-492	2 チャンネル、4-20mA、絶縁型差動入力
750-455、753-455	4 チャンネル、4-20mA、シングルエンド
AI 0-1A	
750-475、753-475	2 チャンネル、0-1A AC/DC、差動入力
AI 0-5A	
750-475/020-000、 753-475/020-000	2 チャンネル、0-5A AC/DC、差動入力
AI 0-10V	
750-467、753-467	2 チャンネル、DC0-10V、シングルエンド
750-477、753-477	2 チャンネル、AC/DC0-10V、差動入力
750-478、753-478	2 チャンネル、DC0-10V、シングルエンド
750-459、753-459	4 チャンネル、DC0-10V、シングルエンド
750-468	4 チャンネル、DC0-10V、シングルエンド
AI DC ±10V	
750-456、753-456	2 チャンネル、DC±10V、差動入力
750-479、753-479	2 チャンネル、DC±10V、差動計測入力
750-476、753-476	2 チャンネル、DC±10V、シングルエンド
750-457、753-457	4 チャンネル、DC±10V、シングルエンド
AI DC 0-30V	
750-483、753-483	2 チャンネル、DC 0-30V、差動計測入力
AI 抵抗センサ	
750-461、753-461	2 チャンネル、抵抗センサ、PT100/RTD
750-481/003-000	2 チャンネル、抵抗センサ、PT100/RTD、EExi
750-460	4 チャンネル、抵抗センサ、PT100/RTD
AI 熱電対	
750-469、753-469	2 チャンネル、診断付 熱電対タイプ：J、K、B、E、N、R、S、T、U、L
AI その他	
750-491	1 チャンネル、抵抗ブリッジ（ひずみゲージ）

5.1.4 アナログ出力モジュール

AO 0-20mA	
750-552、753-552	2 チャンネル、0-20mA
750-585	2 チャンネル、0-20mA、E Exi
750-553、753-553	4 チャンネル、0-20mA
AO 4-20mA	
750-554、753-554	2 チャンネル、4-20mA
750-555、753-555	4 チャンネル、4-20mA
AO DC0-10V	
750-550、753-550	2 チャンネル、DC0-10V
750-560	2 チャンネル、DC0-10V、10 ビット、100mW、24V
750-559、753-559	4 チャンネル、DC0-10V
AO DC $\pm 10V$	
750-556、753-556	2 チャンネル、DC $\pm 10V$
750-557、753-557	4 チャンネル、DC $\pm 10V$

5.1.5 特殊モジュール

カウンタモジュール	
750-404、753-404	アップダウンカウンタ、DC24V、100kHz
750-638、753-638	2 チャンネル、アップダウンカウンタ、DC24V、16 ビット、500Hz
周波数測定	
750-404/000-003	周波数測定
753-404/000-003	
パルス幅出力モジュール	
750-511	2 チャンネルパルス幅出力モジュール、DC24V、短絡保護付、正方向スイッチ
距離/角度測定モジュール	
750-630	SSI トランスミッタインタフェース
750-631/000-004	インクリメンタルエンコーダ・インタフェース、16 ビット
750-635、753-635	デジタルインパルスインタフェース
750-637	インクリメンタルエンコーダ・インタフェース、32 ビット
シリアルインタフェース	
750-650、753-650	RS232C シリアルインタフェース
750-653、753-653	RS485 シリアルインタフェース
750-651	カレントループインタフェース
750-654	データ交換モジュール
DALI/DSI マスタモジュール	
750-641	DALI/DSI マスタモジュール
AS-Interface マスタモジュール	
750-655	AS-Interface マスタモジュール
振動モニタリング	
750-645	2 チャンネル振動速度/ベアリング状態監視 VIB I/O
RTC モジュール	
750-640	RTC モジュール

5.1.6 システムモジュール

内部バス拡張モジュール	
750-627	内部バス拡張、終端モジュール
750-628	内部バス拡張、カプラモジュール
DC 24V 電源入力モジュール	
750-602	DC24V、受動
750-601	DC24V、最大 6.3A、診断なし、ヒューズホルダ付
750-610	DC24V、最大 6.3A、診断付、ヒューズホルダ付
750-625	DC24V、EExi、ヒューズホルダ付
DC 24V 電源入力モジュール、システム電源供給付	
750-613	システム電源、DC24V
AC 120V 電源入力モジュール	
750-615	AC120V、最大 6.3A、診断なし、ヒューズホルダ付
AC 230V 電源入力モジュール	
750-612	AC/DC230V、診断なし、受動
750-609	AC230V、最大 6.3A、診断なし、ヒューズホルダ付
750-611	AC230V、最大 6.3A、診断付、ヒューズホルダ付
フィルタモジュール	
750-624	フィルタモジュール、フィールド機器電源用
750-626	フィルタモジュール、システム/フィールド機器電源用
電源端子拡張モジュール	
750-603、753-603	電源端子拡張モジュール、DC24V
750-604、753-604	電源端子拡張モジュール、DC0V
750-614、753-614	電源端子拡張モジュール、AC/DC 0～230V
分離モジュール	
750-616	分離モジュール
750-621	分離モジュール、電源ジャンパ接点付
バイナリスペーサモジュール	
750-622	バイナリスペーサモジュール
終端モジュール	
750-600	終端モジュール、内部バスのループ

5.2 CANopen 用プロセスデータ構造

I/O モジュールによっては、プロセスデータの構造がフィールドバス固有のものがあります。

CANopen バスカプラ/コントローラの場合、プロセスイメージとしてはバイト構造（ワード配列なし）を用います。1 バイト以上のデータの内部マッピング方法は、インテルフォーマットに従っています。

以降の節には、CANopen バスカプラ/コントローラを用いたときに 750 や 753 シリーズ I/O モジュールをサポートする、全てのバスカプラ/コントローラのプロセスイメージを記述しています。



注意

フィールドバスノード内の I/O モジュールの特定の場所に基づいて、プロセスデータマップ内の位置を決定するために、全ての前方にあるバイトまたはビット単位モジュールのプロセスイメージを考慮しなければなりません。

CANopen オブジェクトディクショナリにおいて、全入力プロセスイメージはインデックス 0x5000 によって読むことができ、また全出力プロセスイメージはインデックス 0x5001 で書くことができます。

5.2.1 デジタル入力モジュール

デジタル入力モジュールは、対応するチャンネルの信号状態を規定するために、チャンネルあたり 1 ビットのデータを提供します。

アナログ入力モジュールもノードに存在した場合、デジタルデータはバイト単位でグループ化し、入力プロセスイメージの中で常にアナログデータの後に付加されます。従って、各バイトに対しサブインデックスが 1 個占有されます。

デジタルモジュールの中には、入力プロセスイメージにおいてチャンネル毎に追加の診断ビットを有するものがあります。診断ビットは発生したエラーを検出するのに用いられます（例：断線や短絡など）。

1 チャンネルデジタル入力モジュール、診断付

750-435

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						診断ビット S 1	データビット DI 1

デジタル入力に対し、オブジェクト 0x6000（0x2000 も可）が使用されます。

2 チャンネルデジタル入力モジュール

750-400, -401, -405, -406, -410, -411, -412, -427, -438, (および全ての枝番),
753-400, -401, -405, -406, -410, -411, -412, -427

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						データビット DI 2 チャンネル 2	データビット DI 1 チャンネル 1

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

2 チャンネルデジタル入力モジュール、診断付

750-421, -424, -425, 753-421, -424, -425

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				診断ビット S 2 チャンネル 2	診断ビット S 1 チャンネル 1	データビット DI 2 チャンネル 2	データビット DI 1 チャンネル 1

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

2 チャンネルデジタル入力モジュール、診断および出力プロセスデータ付

750-418, 753-418

750-418, 753-418 デジタル入力モジュールは、各入力チャンネルに対して診断および肯定応答ビットがあります。エラーが起きた場合、診断ビットがセットされます。エラーが解除された後入力を再開するために、肯定応答ビットをセットしなければなりません。診断データおよび入力データビットは入力プロセスイメージにマッピングされ、また肯定応答ビットは出力プロセスイメージにマッピングされます。

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				診断ビット S 2 チャンネル 2	診断ビット S 1 チャンネル 1	データビット DI 2 チャンネル 2	データビット DI 1 チャンネル 1

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

出力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				肯定応答ビット Q 2 チャンネル 2	肯定応答ビット Q 1 チャンネル 1	0	0

デジタル出力に対し、オブジェクト 0x6200 (0x2100 も可) が使用されます。

4 チャンネルデジタル入力モジュール

750-402, -403, -408, -409, -414, -415, -422, -423, -428, -432, -433,
753-402, -403, -408, -409, -415, -422, -423, -428, -432, -433, -440

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				データビット DI 4 チャンネル 4	データビット DI 3 チャンネル 3	データビット DI 2 チャンネル 2	データビット DI 1 チャンネル 1

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

8 チャンネルデジタル入力モジュール

750-430, -431, -436, -437, 753-430, -431, -434

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
データビット DI 8 チャンネル 8	データビット DI 7 チャンネル 7	データビット DI 6 チャンネル 6	データビット DI 5 チャンネル 5	データビット DI 4 チャンネル 4	データビット DI 3 チャンネル 3	データビット DI 2 チャンネル 2	データビット DI 1 チャンネル 1

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

5.2.2 デジタル出力モジュール

デジタル出力モジュールは対応するチャンネルの出力を制御するために、チャンネルあたり 1 ビットのデータを使用します。これらのビットは出力プロセスイメージにマッピングされます。

アナログ出力モジュールもノードに存在した場合、デジタルデータはバイト単位でグループ化され、出力プロセスイメージの中で常にアナログデータの後に付加されます。従って、各バイトに対しサブインデックスが 1 個占有されます。

1 チャンネルデジタル出力モジュール、入力プロセスデータ付

750-523

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						未使用	ステータス ビット "手動操作"

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

出力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						未使用	制御 DO 1 チャンネル 1

デジタル出力に対し、オブジェクト 0x6200 (0x2100 も可) が使用されます。

2 チャンネルデジタル出力モジュール、

750-501, -502, -509, -512, -513, -514, -517, -535, (および全ての枝番),
753-501, -502, -509, -512, -513, -514, -517

出力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						制御 DO 2 チャンネル 2	制御 DO 1 チャンネル 1

デジタル出力に対し、オブジェクト 0x6200 (0x2100 も可) が使用されます。

2 チャンネルデジタル出力モジュール、診断および入力プロセスデータ付

750-508, -522, 753-507

750-508, -522 および 753-507 デジタル出力モジュールは、各出力チャンネルに対して診断ビットがあります。出力でのエラー（例：過負荷、短絡、断線など）が起きた場合、診断ビットがセットされます。診断データは入力プロセスイメージにマッピングされ、また出力制御ビットは出力プロセスイメージにマッピングされます。

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						診断ビット S2 チャンネル 2	診断ビット S1 チャンネル 1

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

出力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						制御 DO 2 チャンネル 2	制御 DO 1 チャンネル 1

デジタル出力に対し、オブジェクト 0x6200 (0x2100 も可) が使用されます。

750-506, 753-506

750-506, 753-506 デジタル出力モジュールは、各出力チャンネルに対して 2 ビットの診断情報があります。2 ビットの診断情報は、モジュールの正確なエラー状態（例：過負荷、短絡、断線など）を判断するために、チャンネル単位に分解することができます。4 ビットの診断データは入力プロセスイメージにマッピングされ、また出力制御ビットは出力プロセスイメージにマッピングされます。

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				診断ビット S3 チャンネル 4	診断ビット S2 チャンネル 3	診断ビット S1 チャンネル 2	診断ビット S0 チャンネル 1

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

出力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				未使用	未使用	制御 DO 2 チャンネル 2	制御 DO 1 チャンネル 1

デジタル出力に対し、オブジェクト 0x6200 (0x2100 も可) が使用されます。

4 チャンネルデジタル出力モジュール

750-504, -516, -519, -531, 753-504, -516, -531, -540

出力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				制御 DO 4 チャンネル 4	制御 DO 3 チャンネル 3	制御 DO 2 チャンネル 2	制御 DO 1 チャンネル 1

デジタル出力に対し、オブジェクト 0x6200 (0x2100 も可) が使用されます。

4 チャンネルデジタル出力モジュール、診断および入力プロセスデータ付

750-532

750-532 デジタル出力モジュールは、各出力チャンネルに対し診断ビットがあります。出力でのエラー（例：過負荷、短絡、断線など）が起きた場合、診断ビットがセットされます。診断データは入力プロセスイメージにマッピングされ、また出力制御ビットは出力プロセスイメージにマッピングされます。

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				診断ビット S3 チャンネル 4	診断ビット S2 チャンネル 3	診断ビット S1 チャンネル 2	診断ビット S0 チャンネル 1

診断ビット S='0' エラーなし

診断ビット S='1' 過負荷、短絡、断線

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

出力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				制御 DO 4 チャンネル 4	制御 DO 3 チャンネル 3	制御 DO 2 チャンネル 2	制御 DO 1 チャンネル 1

デジタル出力に対し、オブジェクト 0x6200 (0x2100 も可) が使用されます。

8 チャンネルデジタル出力モジュール

750-530, -536, 753-530, -534

出力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
制御 DO 8 チャンネル 8	制御 DO 7 チャンネル 7	制御 DO 6 チャンネル 6	制御 DO 5 チャンネル 5	制御 DO 4 チャンネル 4	制御 DO 3 チャンネル 3	制御 DO 2 チャンネル 2	制御 DO 1 チャンネル 1

デジタル出力に対し、オブジェクト 0x6200 (0x2100 も可) が使用されます。

8 チャンネルデジタル出力モジュール、診断および入力プロセスデータ付

750-537

750-537 デジタル出力モジュールは、各出力チャンネルに対し診断ビットがあります。出力でのエラー（例：過負荷、短絡、断線など）が起きた場合、診断ビットがセットされます。診断データは入力プロセスイメージにマッピングされ、また出力制御ビットは出力プロセスイメージにマッピングされます。

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
診断 ビット S7 チャンネル 8	診断 ビット S6 チャンネル 7	診断 ビット S5 チャンネル 6	診断 ビット S4 チャンネル 5	診断 ビット S3 チャンネル 4	診断 ビット S2 チャンネル 3	診断 ビット S1 チャンネル 2	診断 ビット S0 チャンネル 1

診断ビット S='0' エラーなし

診断ビット S='1' 過負荷、短絡、断線

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

出力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
制御 DO 8 チャンネル 8	制御 DO 7 チャンネル 7	制御 DO 6 チャンネル 6	制御 DO 5 チャンネル 5	制御 DO 4 チャンネル 4	制御 DO 3 チャンネル 3	制御 DO 2 チャンネル 2	制御 DO 1 チャンネル 1

デジタル出力に対し、オブジェクト 0x6200 (0x2100 も可) が使用されます。

5.2.3 アナログ入力モジュール

アナログ入力モジュールは、チャンネル毎に 16 ビットの測定したアナログデータおよび 8 ビットの制御/ステータスを持っています。しかし CANopen バスカブラ/コントローラは、8 ビットの制御/ステータスにはアクセスしません。従って CANopen バスカブラ/コントローラはチャンネル毎に 16 ビットのアナログデータのみにアクセスできます。このデータはバイト単位でグループ化され、入力プロセスイメージにインテルフォーマットでマッピングされます。

ノード内にデジタル入力モジュールもある場合、アナログ入力データは入力プロセスイメージの中で、常にデジタルデータの前にマッピングされます。

1 チャンネルアナログ入力モジュール

750-491、(および全ての枝番)

入力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	D0	測定値U _D
	1	D1	
n+1	2	D2	測定値U _{ref}
	3	D3	

このモジュールは 2x2 バイトで構成され、2 バイト特殊モジュール入力用にオブジェクト 0x6401 (0x2400 も可) が使用できます。従って各測定値に対してサブインデックス 1 個が占有されます。

2 チャンネルアナログ入力モジュール

750-452, -454, -456, -461, -462, -465, -466, -467, -469, -472, -474, -475, -476, -477, -478, -479, -480, -481, -483, -485, -492, (および全ての枝番), 753-452, -454, -456, -461, -465, -466, -467, -469, -472, -474, -475, -476, -477, -478, -479, -483, -492, (および全ての枝番)

入力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	D0	測定値、チャンネル 1
	1	D1	
n+1	2	D2	測定値、チャンネル 2
	3	D3	

このモジュールは 2x2 バイトで構成され、2 バイト特殊モジュール入力用にオブジェクト 0x6401 (0x2400 も可) が使用できます。従って各測定値に対してサブインデックス 1 個が占有されます。

4 チャンネルアナログ入力モジュール

750-453, -455, -457, -459, -460, -468、(および全ての枝番)、
753-453, -455, -457, -459

入力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	D0	測定値、チャンネル 1
	1	D1	
n+1	2	D2	測定値、チャンネル 2
	3	D3	
n+2	4	D4	測定値、チャンネル 3
	5	D5	
n+3	6	D6	測定値、チャンネル 4
	7	D7	

このモジュールは 4x2 バイトで構成され、2 バイト特殊モジュール入力用にオブジェクト 0x6401 (0x2400 も可) が使用できます。従って各測定値に対してサブインデックス 1 個が占有されます。

5.2.4 アナログ出力モジュール

アナログ出力モジュールは、チャンネル毎に 16 ビットのアナログデータおよび 8 ビットの制御/ステータスを持っています。しかし CANopen バスカプラ/コントローラは、8 ビットの制御/ステータスにはアクセスしません。CANopen ではステータスバイトはエマージェンシ・テレグラムとしてオフにセットされます。従って CANopen バスカプラ/コントローラはチャンネル毎に 16 ビットのアナログデータのみを送出できます。このデータはバイト単位でグループ化され、出力プロセスイメージにインテルフォーマットでマッピングされます。

ノード内にデジタル出力モジュールもある場合、アナログ出力データは出力プロセスイメージの中で、常にデジタルデータの前にマッピングされます。

2 チャンネルアナログ出力モジュール

750-550, -552, -554, -556, -560, -585、(および全ての枝番)、
753-550, -552, -554, -556

出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	D0	出力値、チャンネル 1
	1	D1	
n+1	2	D2	出力値、チャンネル 2
	3	D3	

このモジュールは 2x2 バイトで構成され、2 バイト特殊モジュール出力用にオブジェクト 0x6411 (0x2500 も可) が使用できます。従って各チャンネルに対してサブインデックス 1 個が占有されます。.

4 チャンネルアナログ出力モジュール

750-553, -555, -557, -559, 753-553, -555, -557, -559

出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	D0	出力値、チャンネル 1
	1	D1	
n+1	2	D2	出力値、チャンネル 2
	3	D3	
n+2	4	D4	出力値、チャンネル 3
	5	D5	
n+3	6	D6	出力値、チャンネル 4
	7	D7	

このモジュールは 4x2 バイトで構成され、2 バイト特殊モジュール出力用にオブジェクト 0x6411 (0x2500 も可) が使用できます。従って各チャンネルに対してサブインデックス 1 個が占有されます。.

5.2.5 特殊モジュール

各々のモジュールでは、データバイトの他に制御/ステータスバイトもまたプロセスイメージにマッピングされます。制御/ステータスバイトは、上位の制御システムとモジュールが双方向のデータ交換をするのに必要となります。制御バイトは制御システムからモジュールに、ステータスバイトはモジュールから制御システムに送信されます。例えばこれにより、制御バイトでカウンタを設定するとか、またはステータスバイトで測定範囲の超過や過小を表示することなどができます。



詳細情報

特定のモジュールの制御/ステータスバイトの構造に関する詳細は「I/O モジュール取扱い説明書」をご覧ください。I/O モジュールの詳細に関しては、以下の WEB サイトから入手可能です。

http://www.wago.co.jp/io/11_download_Japanese.htm

カウンタモジュール

750-404、(および/000-005 以外の全ての枝番)、
753-404、(および枝番/000-003)

上記のカウンタモジュールは、入力および出力プロセスイメージの両方に各々5 バイト (カウンタデータ用 4 バイトおよび制御/ステータス用 1 バイト) のユーザデータがあります。カウンタ値は 32 ビットで構成されます。次の表は入力および出力プロセスイメージを表しており、全部で 6 バイトが各プロセスイメージにマッピングされます。

入力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	S	ステータスバイト
	1	-	未使用
	2	D0	カウンタ値
	3	D1	
	4	D2	
	5	D3	

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール入力用にオブジェクト 0x3200 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C	制御バイト
	1	-	未使用
	2	D0	カウンタ設定値
	3	D1	
	4	D2	
	5	D3	

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール出力用にオブジェクト 0x3300 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

750-404 /000-005

上記のカウンタモジュールは、入力および出力プロセスイメージの両方に各々5 バイト（カウンタデータ用 4 バイトおよび制御/ステータス用 1 バイト）のユーザデータがあります。カウンタ値は 16 ビットで構成されます。次の表は入力および出力プロセスイメージを表しており、全部で 6 バイトが各プロセスイメージにマッピングされます。

入力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	S	ステータスバイト
	1	-	未使用
	2	D0	カウンタ値、カウンタ 1
	3	D1	
	4	D2	カウンタ値、カウンタ 2
	5	D3	

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール入力用にオブジェクト 0x3200 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C	制御バイト
	1	-	未使用
	2	D0	カウンタ設定値、カウンタ 1
	3	D1	
	4	D2	カウンタ設定値、カウンタ 2
	5	D3	

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール出力用にオブジェクト 0x3300 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

750-638, 753-638

上記のカウンタモジュールは、入力および出力プロセスイメージの両方に各々6 バイト（カウンタデータ用 4 バイトおよび制御/ステータス用 2 バイト）のユーザデータがあります。2 つのカウンタ値は各々16 ビットで構成されます。次の表は入力および出力プロセスイメージを表しており、全部で 6 バイトが各プロセスイメージにマッピングされます。

入力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	S0	ステータスバイト、カウンタ 1
	1	D0	カウンタ値、カウンタ 1
	2	D1	
n+1	3	S1	ステータスバイト、カウンタ 2
	4	D2	カウンタ値、カウンタ 2
	5	D3	

このモジュールは 2x3 バイトで表されますので、3 バイト特殊モジュール入力用にオブジェクト 0x2600 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C0	制御バイト、カウンタ 1
	1	D0	カウンタ設定値、カウンタ 1
	2	D1	
n+1	3	C1	制御バイト、カウンタ 2
	4	D2	カウンタ設定値、カウンタ 2
	5	D3	

このモジュールは 2x3 バイトで表されますので、3 バイト特殊モジュール出力用にオブジェクト 0x2700 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

パルス幅出力モジュール

750-511、(および全ての枝番)

上記のパルス幅出力モジュールは、入力および出力プロセスイメージの両方に合計 6 バイト（チャンネルデータ用 4 バイトおよび制御/ステータス用 2 バイト）のユーザデータがあります。2 つのチャンネル値は各々 16 ビットで構成されます。各チャンネルにはチャンネル毎の制御/ステータスバイトがあります。次の表は入力および出力プロセスイメージを表しており、全部で 6 バイトが各プロセスイメージにマッピングされます。

入力および出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C0/S0	制御/ステータスバイト、チャンネル 1
	1	D0	データ値、チャンネル 1
	2	D1	
n+1	3	C1/S1	制御/ステータスバイト、チャンネル 2
	4	D2	データ値、チャンネル 2
	5	D3	

このモジュールは 2x3 バイトで表されますので、3 バイト特殊モジュール入力用にオブジェクト 0x2600 が、また 3 バイト特殊モジュール出力用にオブジェクト 0x2700 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

シリアルインタフェースモジュール、代替データフォーマット付

750-650、(および全ての枝番/000-002, -004, -006, -009, -010, -011, -012, -013)

750-651、(および全ての枝番/000-002, -003)

750-653、(および全ての枝番/000-002, -007)

**メモ**

シリアルインタフェースモジュールの中で、自由にパラメータが設定できる枝番/003-000 付のものは、目的とする操作モードを設定することができます。このモードに基づいて、モジュールのプロセスイメージは対応した枝番のものと同じです。

上記の代替データフォーマット付シリアルインタフェースモジュールは、入力および出力プロセスイメージの両方に合計 4 バイトのユーザデータ（シリアルデータ用 3 バイトおよび制御/ステータス用 1 バイト）があります。次の表は入力および出力プロセスイメージを表しており、全部で 4 バイトが各プロセスイメージにマッピングされます

入力および出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C/S	制御/ステータスバイト
	1	D0	データバイト
n+1	2	D1	
	3	D2	

このモジュールは 2x2 バイトで表されますので、2 バイト特殊モジュール入力用にオブジェクト 0x6401（0x2400 も可）が、また 2 バイト特殊モジュール出力用にオブジェクト 0x6411（0x2500 も可）が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

シリアルインタフェースモジュール、標準データフォーマット付

750-650/000-001, -014, -015, -016

750-651/000-001

750-653/000-001, -006

上記の標準データフォーマット付シリアルインタフェースモジュールは、入力および出力プロセスイメージの両方に合計 6 バイトのユーザデータ（シリアルデータ用 5 バイトおよび制御/ステータス用 1 バイト）があります。次の表は入力および出力プロセスイメージを表しており、全部で 6 バイトが各プロセスイメージにマッピングされます

入力および出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C/S	制御/ステータスバイト
	1	D0	データバイト
	2	D1	
	3	D2	
	4	D3	
	5	D4	

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール入力用にオブジェクト 0x3200 が、また 6 バイト特殊モジュール出力用にオブジェクト 0x3300 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

データ交換モジュール

750-654、(および枝番/000-001)

データ交換モジュールは、入力および出力プロセスイメージの両方に合計 4 バイトのユーザデータがあります。次の表は入力および出力プロセスイメージを表しており、全部で 4 バイトが各プロセスイメージにマッピングされます

入力および出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	D0	データバイト
	1	D1	
n+1	2	D2	
	3	D3	

このモジュールは 2x2 バイトで表されますので、2 バイト特殊モジュール入力用にオブジェクト 0x6401 (0x2400 も可) が、また 2 バイト特殊モジュール出力用にオブジェクト 0x6411 (0x2500 も可) が使用されます。従って各モジュールに対して、2 個のサブインデックスを占有します。

SSI トランスミッタインタフェースモジュール、代替データフォーマット付

750-630、(および枝番/000-001, -002, -006, -008, -009, -011, -012, -013)



メモ

SSI トランスミッタインタフェースモジュールの中で自由にパラメータが設定できる枝番/003-000 付のものは、目的の操作モードを設定することができます。このモードに基づいて、モジュールのプロセスイメージは対応した枝番のものと同じです。

上記の代替データフォーマット付 SSI トランスミッタインタフェースモジュールは、入力プロセスイメージに合計 4 バイトのユーザデータがあります。次の表は入力プロセスイメージを表しており、全部で 4 バイトがプロセスイメージにマッピングされます

入力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	D0	データバイト
	1	D1	
n+1	2	D2	
	3	D3	

このモジュールは 2x2 バイトで表されますので、2 バイト特殊モジュール入力用にオブジェクト 0x6401 (0x2400 も可) が使用されます。従って各モジュールに対して、2 個のサブインデックスを占有します。

SSI トランスミッタインタフェースモジュール、標準データフォーマット付

750-630/000-004, -005, -007

上記の標準データフォーマット付 SSI トランスミッタインタフェースモジュールは、入力プロセスイメージに合計 5 バイトのユーザデータ（ユーザデータ用 4 バイトおよびステータス用 1 バイト）があります。次の表は入力プロセスイメージを表しており、全部で 6 バイトがプロセスイメージにマッピングされます。

入力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	S	ステータスバイト
	1	-	未使用
	2	D0	データバイト
	3	D1	
	4	D2	
	5	D3	

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール入力用にオブジェクト 0x3200 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

インクリメンタルエンコーダ・インタフェースモジュール

750-631

このインクリメンタルエンコーダ・インタフェースモジュールは、5 バイトの入力データと 3 バイトの出力データを持っています。次の表は入力および出力プロセスイメージを表しており、各 6 バイトが各々のプロセスイメージにマッピングされます。

入力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	S	ステータスバイト
	1	D0	カウンタワード
	2	D1	
	3	-	未使用
	4	D2	ラッチワード
	5	D3	

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール入力用にオブジェクト 0x3200 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C	制御バイト
	1	D0	カウンタ設定ワード
	2	D1	
	3	-	未使用
	4	-	
	5	-	

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール出力用にオブジェクト 0x3300 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

750-637

このインクリメンタルエンコーダ・インタフェースモジュールは、入力および出力プロセスイメージの両方に合計 6 バイトのユーザデータ（4 バイトのエンコーダデータと 2 バイトの制御/ステータス）を持っています。次の表は入力および出力プロセスイメージを表しており、各 6 バイトが各々のプロセスイメージにマッピングされます。

入力および出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C0/S0	制御/ステータスバイト 1
	1	D0	データ値
	2	D1	
n+1	3	C1/S1	制御/ステータスバイト 2
	4	D2	データ値
	5	D3	

このモジュールは 2x3 バイトで表されますので、3 バイト特殊モジュール入力用にオブジェクト 0x2600 が、また 3 バイト特殊モジュール出力用にオブジェクト 0x2700 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

デジタルインパルスインタフェースモジュール

750-635, 753-635

このデジタルインパルスインタフェースモジュールは、入力および出力プロセスイメージの両方に合計 4 バイトのユーザデータ（3 バイトのモジュールデータと 1 バイトの制御/ステータス）を持っています。次の表は入力および出力プロセスイメージを表しており、各 4 バイトが各々のプロセスイメージにマッピングされます。

入力および出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C0/S0	制御/ステータスバイト
	1	D0	データ値
	2	D1	
	3	D2	

このモジュールは 1x4 バイトで表されますので、4 バイト特殊モジュール入力用にオブジェクト 0x2800 が、また 4 バイト特殊モジュール出力用にオブジェクト 0x2900 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

RTC モジュール

750-640

この RTC モジュールは、入力および出力プロセスイメージの両方に合計 6 バイトのユーザデータ（4 バイトのモジュールデータ、1 バイトの制御/ステータスおよび 1 バイトのコマンド用 ID）を持っています。次の表は入力および出力プロセスイメージを表しており、各 6 バイトが各々のプロセスイメージにマッピングされます。

入力および出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C/S	制御/ステータスバイト
	1	ID	コマンドバイト
	2	D0	データバイト
	3	D1	
	4	D2	
	5	D3	

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール入力用にオブジェクト 0x3200 が、また 6 バイト特殊モジュール出力用にオブジェクト 0x3300 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

DALI/DSI マスタモジュール

750-641

この DALI/DSI マスタモジュールは、入力および出力プロセスイメージの両方に合計 6 バイトのユーザデータ（5 バイトのモジュールデータおよび 1 バイトの制御/ステータス）を持っています。次の表は入力および出力プロセスイメージを表しており、各 6 バイトが各々のプロセスイメージにマッピングされます。

入力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	S	ステータスバイト
	1	D0	DALI 応答
	2	D1	DALI アドレス
	3	D2	メッセージ 3
	4	D3	メッセージ 2
	5	D4	メッセージ 1

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール入力用にオブジェクト 0x3200 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C	制御バイト
	1	D0	DALI コマンド、DSI 調光値
	2	D1	DALI アドレス
	3	D2	パラメータ 2
	4	D3	パラメータ 1
	5	D4	コマンド拡張

このモジュールは 1x6 バイトで表されますので、6 バイト特殊モジュール出力用にオブジェクト 0x3300 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

振動速度/ベアリング状態監視 VIB I/O

750-645

この振動速度/ベアリング状態監視 VIB I/O は、入力および出力プロセスイメージの両方に合計 12 バイトのユーザデータ（8 バイトのモジュールデータと 4 バイトの制御/ステータス）を持っています。次の表は入力および出力プロセスイメージを表しており、各 12 バイトが各々のプロセスイメージにマッピングされます。

入力および出力プロセスイメージ			
サブインデックス	オフセット	バイト名	意味
n	0	C0/S0	制御/ステータスバイト (論理チャンネル 1、センサ入力 1)
	1	D0	データバイト (論理チャンネル 1、センサ入力 1)
	2	D1	
n+1	3	C1/S1	制御/ステータスバイト (論理チャンネル 2、センサ入力 2)
	4	D2	データバイト (論理チャンネル 2、センサ入力 2)
	5	D3	
n+2	6	C2/S2	制御/ステータスバイト (論理チャンネル 3、センサ入力 1)
	7	D4	データバイト (論理チャンネル 3、センサ入力 1)
	8	D5	
n+3	9	C3/S3	制御/ステータスバイト (論理チャンネル 4、センサ入力 2)
	10	D6	データバイト (論理チャンネル 4、センサ入力 2)
	11	D7	

このモジュールは 4x3 バイトで表されますので、3 バイト特殊モジュール入力用にオブジェクト 0x2600 が、また 3 バイト特殊モジュール出力用にオブジェクト 0x2700 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。

AS-Interface マスタモジュール

750-655

AS-Interface マスタのプロセスイメージの最大値は 48 バイトからなります。CANopen バスコントローラを使用したときは、メールボックスの最大長は 6 バイトに制限されます。各 PDO は 8 バイトのデータを持つことができます。

AS-Interface マスタモジュールに割り当てられる最初の PDO は、制御/ステータスバイト、空白 1 バイト、および最大 6 バイトのメールボックスまたはプロセスデータが入ります。後続の PDO は AS-Interface プロセスデータが入ります。

以下の表には、メールボックスが恒久的に重なって使用されたとき、占有 PDO 数に対するプロセスイメージ長の割り当てを示しています（モード 1）。

入力および出力プロセスイメージ						
プロセスイメージ長	12 バイト	20 バイト	24 バイト	32 バイト	40 バイト	48 バイト
n 番目 PDO	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックス	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックス	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックス	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックス	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックス	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックス
n+1 番目 PDO	4 バイトプロ セスデータ (フラグ、ス レーブ 1/1A – スレーブ 7/7A) 4 バイト空白 (予約)	8 バイトプロ セスデータ (フラグ、ス レーブ 1/1A – スレーブ 15/15A)	8 バイトプロ セスデータ (フラグ、ス レーブ 1/1A – スレーブ 15/15A)	8 バイトプロ セスデータ (フラグ、ス レーブ 1/1A – スレーブ 15/15A)	8 バイトプロ セスデータ (フラグ、ス レーブ 1/1A – スレーブ 15/15A)	8 バイトプロ セスデータ (フラグ、ス レーブ 1/1A – スレーブ 15/15A)
n+2 番目 PDO	後続モジュ ール用空領域	4 バイトプロ セスデータ (スレーブ 16/16A – ス レーブ 23/23A) 4 バイト空白 (予約)	8 バイトプロ セスデータ (スレーブ 16/16A – ス レーブ 31/31A)	8 バイトプロ セスデータ (スレーブ 16/16A – ス レーブ 31/31A)	8 バイトプロ セスデータ (スレーブ 16/16A – ス レーブ 31/31A)	8 バイトプロ セスデータ (スレーブ 16/16A – ス レーブ 31/31A)
n+3 番目 PDO		f 後続モジュ ール用空領域	後続モジュ ール用空領域	8 バイトプロ セスデータ (スレーブ 1B – スレー ブ 15B)	8 バイトプロ セスデータ (スレーブ 1B – スレー ブ 15B)	8 バイトプロ セスデータ (スレーブ 1B – スレー ブ 15B)
n+4 番目 PDO				後続モジュ ール用空領域	8 バイトプロ セスデータ (スレーブ 16B – スレー ブ 31B)	8 バイトプロ セスデータ (スレーブ 16B – スレー ブ 31B)
n+5 番目 PDO					後続モジュ ール用空領域	8 バイト空白 (予約)
n+6 番目 PDO						後続モジュ ール用空領域

ここで、n 番目 PDO は AS-Interface マスタモジュールで占有される、最初の PDO を表します。ここにはステータス/制御バイト、空白 1 バイト、最大 6 バイトのメールボックスデータが入ります。

恒久的に重なったメールボックス長が 0 バイトであった場合、n 番目 PDO のみにステータス/制御バイトと空白 1 バイトが入ります。

もし AS-Interface マスタモジュールのプロセスイメージが 12 バイトまたは 20 バイトの大きさならば、最終 PDO は全部は占有されません。この後のモジュールは次の PDO から始まります。



メモ

メールボックスのサイズが 6 バイトで、62 台の AS-Interface スレーブが搭載されているとき、プロセスイメージのサイズが 40 バイトであれば、プロセスデータ全てを送信するには十分です。このときモジュールは 5 個の PDO を占有します。

次に示すように、占有 PDO 数に対するプロセスイメージ長の割り当ては、オーバーラップ可メールボックスを伴った操作モードで適用されます（モード 2）。

入力および出力プロセスイメージ						
プロセスイメージ長	12 バイト	20 バイト	24 バイト	32 バイト	40 バイト	48 バイト
n 番目 PDO	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックスまたは 6 バイト プロセスデータ (フラグ、 スレーブ 1/1A -スレーブ 11/11A)	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックスまたは 6 バイト プロセスデータ (フラグ、 スレーブ 1/1A -スレーブ 11/11A)	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックスまたは 6 バイト プロセスデータ (フラグ、 スレーブ 1/1A -スレーブ 11/11A)	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックスまたは 6 バイト プロセスデータ (フラグ、 スレーブ 1/1A -スレーブ 11/11A)	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックスまたは 6 バイト プロセスデータ (フラグ、 スレーブ 1/1A -スレーブ 11/11A)	1 ステータス/ 制御バイト 1 空白バイト 6 バイトメール ボックスまたは 6 バイト プロセスデータ (フラグ、 スレーブ 1/1A -スレーブ 11/11A)
n+1 番目 PDO	4 バイトプロ セスデータ (スレーブ 12/12A - ス レーブ 19/19A) 4 バイト空白 (予約)	8 バイトプロ セスデータ (スレーブ 12/12A - ス レーブ 27/27A)	8 バイトプロ セスデータ (スレーブ 12/12A - ス レーブ 27/27A)	8 バイトプロ セスデータ (スレーブ 12/12A - ス レーブ 27/27A)	8 バイトプロ セスデータ (スレーブ 12/12A - ス レーブ 27/27A)	8 バイトプロ セスデータ (スレーブ 12/12A - ス レーブ 27/27A)
n+2 番目 PDO	後続モジュール 用空領域	4 バイトプロ セスデータ (スレーブ 28/28A - ス レーブ, 3B) 4 バイト空白 (予約)	8 バイトプロ セスデータ (スレーブ 28/28A - ス レーブ 11B)	8 バイトプロ セスデータ (スレーブ 28/28A - ス レーブ 11B)	8 バイトプロ セスデータ (スレーブ 28/28A - ス レーブ 11B)	8 バイトプロ セスデータ (スレーブ 28/28A - ス レーブ 11B)
n+3 番目 PDO		後続モジュール 用空領域	後続モジュール 用空領域	8 バイトプロ セスデータ (スレーブ 12B - スレー ブ 27B)	8 バイトプロ セスデータ (スレーブ 12B - スレー ブ 27B)	8 バイトプロ セスデータ (スレーブ 12B - スレー ブ 27B)
n+4 番目 PDO				後続モジュール 用空領域	2 バイトプロ セスデータ (スレーブ 28B - スレー ブ 31B) 6 バイト空白 (予約)	8 バイトプロ セスデータ (スレーブ 28B - スレー ブ 31B) 6 バイト空白 (予約)
n+5 番目 PDO					後続モジュール 用空領域	8 バイト空白 (予約)
n+6 番目 PDO						後続モジュール 用空領域

n 番目 PDO はメールボックスがオーバーラップされたとき、ステータス/制御バイト、空白 1 バイトおよび 6 バイトのメールボックスデータが入るか、またはプロセスデータの最初の 6 バイトが入ります。後続の PDO は追加のプロセスデータを含みます。



メモ

メールボックスがオーバーラップされるとき、プロセスデータの最初の 6 バイトにアクセスすることはできません (フラグおよびスレーブ 1/1A~11/11A)。

AS-Interface マスタモジュールのプロセスイメージが 12、20、40 または 48 バイト長の場合、最終 PDO は完全に占有はされません。このとき追加のモジュールは、次の PDO からスタートします。

このモジュールは 1x12～48 バイトで表されますので、ゲートウェイモジュール入力用にオブジェクト 0x4200～0x4202 が、またゲートウェイモジュール出力用にオブジェクト 0x4300～4302 が使用されます。従って各モジュールに対して、1 個のサブインデックスを占有します。サブインデックス 1 にはメールボックス長、サブインデックス 2 にはメールボックス、サブインデックス 3～48 にはプロセスデータが入ります。

5.2.6 システムモジュール

診断付システムモジュール

750-610, -611

750-610 および 750-611 電源入力モジュールは、内部電源の監視用に入力プロセスイメージに 2 ビットの診断ビットを備えています。

入力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						診断ビット S2 ヒューズ	診断ビット S1 電圧

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。

バイナリスペーサモジュール

750-622

バイナリスペーサモジュール 750-622 は 2 チャンネルデジタル入力モジュールまたは出力モジュールの代替のように動作し、設定条件を選択することにより、チャンネルあたり 1、2、3 または 4 ビットを占めます。これに従って、入力または出力プロセスイメージのどちらかにおいて、2、4、6 または 8 ビットが占有されます。

入力および出力プロセスイメージ							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(データビット DI 8)	(データビット DI 7)	(データビット DI 6)	(データビット DI 5)	(データビット DI 4)	(データビット DI 3)	(データビット DI 2)	(データビット DI 1)

デジタル入力に対し、オブジェクト 0x6000 (0x2000 も可) が使用されます。デジタル出力に対し、オブジェクト 0x6200 (0x2100 も可) が使用されます。

6 爆発性環境での使用について

6.1 はじめに

多くの化学会社または石油化学会社では、製造工場、製造機械、およびプロセス自動化機械などにおいて、爆発性のガス - 空気、蒸気 - 空気、ダスト - 空気などの混合気を使用しています。したがって、その種の工場やシステムで使用する電気機器では、傷害や装置および設備の損傷につながる爆発の危険性を解消する必要があります。このことは、国内外を問わず法律、指令、あるいは規制という形で実行されています。WAGO-I/O-SYSTEM 750（電気機器）は、ゾーン 2（2 種危険度）の爆発性環境で使用するよう設計されています。防爆に関する基本条件は、以下のように規定されています。

6.2 保護対策

爆発性雰囲気形成を防止する方法について最初に説明します。ごく一例を挙げれば、可燃性液体の使用を避ける、濃度を下げる、換気するなどの方法があります。しかし、基本的な防爆対策が実施できない状況も多数存在します。そのような場合には二次的な防爆対策を実施します。この二次対策の詳細について以下に説明します。

6.3 CENELEC および IEC に基づく分類

ここに記載する仕様は欧州で有効なものであり、CENELEC（欧州電気技術標準化委員会）の EN50xxx に準じています。なお、その国際規格版は IEC（国際電気標準会議 : International Electrotechnical Commission）の IEC 60079-x です。

6.3.1 区 分

爆発性環境とは、雰囲気が爆発性を帯びる恐れのある場所を指します。爆発性とは、空気中を漂うガス、フューム、ミスト、ダストなどの形で存在し、ある許容温度を超えて加熱されるか、またはアーク放電や火花にさらされると、爆発する恐れがある可燃性物質の混合気を意味します。爆発性雰囲気の濃度レベルを表すために爆発危険ゾーンという指標が作られています。この区分は、爆発が発生する可能性に基づいており、技術安全および実現可能性の両面から極めて重要です。爆発性環境で恒常的に使用される電気機器には、危険な爆発性環境にほとんどまたは短期間しか置かれない電気機器よりも厳しい条件が課されます。

ガス、フェーム、ミストによる爆発性環境：

- ゾーン 0 領域は、爆発性雰囲気は常時または長期間（>1000 時間／年）さらされる環境です。
- ゾーン 1 領域は、爆発性雰囲気が時折発生する（10～1000 時間／年）と予想される環境です。
- ゾーン 2 領域は、爆発性雰囲気がほとんどまたは短期間しか発生しない（<10 時間／年）と予想される環境です。

浮遊ダストが存在する爆発性環境：

- ゾーン 20 領域は、爆発性雰囲気は常時または長期間（>1000 時間／年）さらされる環境です。
- ゾーン 21 領域は、爆発性雰囲気が時折発生する（10～1000 時間／年）と予想される環境です。
- ゾーン 22 領域は、爆発性雰囲気がほとんどまたは短期間しか発生しない（<10 時間／年）と予想される環境です。

6.3.2 防爆グループ

この他、爆発性環境で使用される電気機器は以下に示す 2 つのグループに分類されます。

グループ I： グループ I に分類されるのは、可燃性ガスが存在する採鉱現場で使用される電気機器です。

グループ II： グループ II に分類されるのは、上記以外の爆発性環境で使用される電気機器です。グループ II は環境中に存在するガスの種類によってさらに IIA、IIB、および IIC に分類されます。このサブグループでは、物質／ガスの種類によって着火エネルギー特性が異なることが考慮されています。したがって、このサブグループには、代表的なガスの種類が以下のように指定されています。

- IIA：プロパン
- IIB：エチレン
- IIC：水素

代表的なガスの種類における最小着火エネルギー				
防爆グループ	I	IIA	IIB	IIC
ガス	メタン	プロパン	エチレン	水素
着火エネルギー（μJ）	280	250	82	16

水素は多くの化学工場で使用されるため、IIC の防爆グループに対して最高の安全対策が求められるケースがよくあります。

6.3.3 装置カテゴリー

さらに、使用領域（ゾーン）と使用条件（防爆グループ）の組み合わせは、電氣的運転手段によって以下のように分類されます。

装置 カテゴリー	防爆 グループ	使用領域
M1	I	可燃性ガスの防爆
M2	I	可燃性ガスの防爆
1G	II	ガス、フューム、ミストによるゾーン 0 の爆発性環境
2G	II	ガス、フューム、ミストによるゾーン 1 の爆発性環境
3G	II	ガス、フューム、ミストによるゾーン 2 の爆発性環境
1D	II	ダストによるゾーン 20 の爆発性環境
2D	II	ダストによるゾーン 21 の爆発性環境
3D	II	ダストによるゾーン 22 の爆発性環境

6.3.4 温度等級

防爆グループ I の電気機器における最高表面温度は 150℃（危険要因が炭塵堆積の場合）または 450℃（炭塵堆積の危険がない場合）です。

防爆グループ II の電気機器については、すべての着火防止タイプに対する最高表面温度により、以下に示す温度等級に分類されます。表に示した温度は、電気機器の動作および試験を 40℃の周囲温度で行った場合の値です。存在する爆発性雰囲気中の最低着火温度は、最大表面温度よりも高くなければなりません。

温度等級	最高表面温度	可燃性物質の着火温度
T1	450℃	450℃超
T2	300℃	300℃～450℃超
T3	200℃	200℃～300℃超
T4	135℃	135℃～200℃超
T5	100℃	100℃～135℃超
T6	85℃	85℃～100℃超

各温度等級および防爆グループに含まれる物質の割合を以下の表に示します。

温度等級						
T1	T2	T3	T4	T5	T6	合計*
26.6 %	42.8 %	25.5 %				
94.9 %			4.9 %	0 %	0.2 %	432
防爆グループ						
IIA	IIB	IIC				合計*
85.2 %	13.8 %	1.0 %				501

*分類した物質の数

6.3.5 着火保護のタイプ

着火保護は、周囲の爆発性雰囲気への着火を防止するために電子機器に施すべき特別な対策を規定します。このため、着火保護は、以下のように区別されます。

識別	CENELEC 規格	IEC 規格	内 容	適 用
EEx o	EN 50 015	IEC 79	油入防爆	ゾーン 1+2
EEx p	EN 50 016	IEC 79	内圧防爆	ゾーン 1+2
EEx q	EN 50 017	IEC 79	砂詰め防爆	ゾーン 1+2
EEx d	EN 50 018	IEC 79	耐圧防爆	ゾーン 1+2
EEx e	EN 50 019	IEC 79	安全増防爆	ゾーン 1+2
EEx m	EN 50 028	IEC 79	モールド防爆	ゾーン 1+2
EEx I	EN 50 020 (ユニット) EN 50 039 (システム)	IEC 79	本質安全防爆	ゾーン 0+1+2
EEx n	EN 50 021	IEC 79	ゾーン 2 用の電気機器 (下記参照)	ゾーン 2

「n」タイプの着火防止は、ゾーン 2 の防爆電気機器についてのみ使用されます。ゾーン 2 とは、爆発性雰囲気がほとんどまたは短期間しか発生しないと予想される環境です。これは、防爆構造が必要なゾーン 1 と、溶接作業が常に許容されるような安全領域との中間的な領域です。

このような電気機器を対象とする規格は世界規模で策定されつつあります。EN 50 021 規格では、電気機器メーカーは、たとえばオランダの KEMA やドイツの PTB などの所轄機関から検査機器が上述の規格草案を満たすことを証明する合格証が得られるようになっています。また「n」タイプの着火保護では、電気機器に次の拡張ラベリングを含めたラベリングを行う必要があります。

- A：スパークを発生しない（リレーもスイッチもない機能モジュール）
- AC：スパークを発生するが接点がシールで保護されている（リレーはあるがスイッチはない機能モジュール）
- L：制限されたエネルギー（スイッチを備えた機能モジュール）



詳細情報

詳細については国内および／または国際的な規格、指令、および規則を参照してください。

6.4 NEC 500 に基づく分類

北米では NEC 500（NEC=米国電気規格）に基づく次のような分類が使用されます。

6.4.1 区 分

区分（Division）は、危険な状態のタイプに関わらず、危険な状態が発生する確率の高さを示します。

可燃性ガス、フューム、ミスト、ダストによって爆発の危険性がある場所	
区分 1	爆発性雰囲気の時折（10～1000 時間/年）または常時または長期間発生する（>1000 時間/年）と予想される環境
区分 2	爆発性雰囲気がほとんどまたは短期間しか発生しない（<10 時間/年）と予想される環境

6.4.2 防爆グループ

爆発の危険性がある場所で使用する電気機器は、次の 3 つの危険カテゴリーに分類されます。

クラス I（ガスおよびフューム）	グループ A（アセチレン） グループ B（水素） グループ C（エチレン） グループ D（メタン）
クラス II（ダスト）	グループ E（金属粉末） グループ F（炭塵） グループ G（小麦粉、澱粉、穀物粉末）
クラス III（繊維）	サブグループなし

6.4.3 温度等級

爆発の危険性がある場所で使用する電気機器は、以下の温度等級に分類されます。

温度等級	最高表面温度	可燃性物質の着火温度
T1	450℃	450℃超
T2	300℃	300℃～450℃超
T2A	280℃	280℃～300℃超
T2B	260℃	260℃～280℃超
T2C	230℃	230℃～260℃超
T2D	215℃	215℃～230℃超
T3	200℃	200℃～215℃超
T3A	180℃	180℃～200℃超
T3B	165℃	165℃～180℃超
T3C	160℃	160℃～165℃超
T4	135℃	135℃～160℃超
T4A	120℃	120℃～135℃超
T5	100℃	100℃～120℃超
T6	85℃	85℃～100℃超

6.5 識別（ラベリング）

6.5.1 欧州

CENELEC および IEC によるラベリングの例を以下に示します。

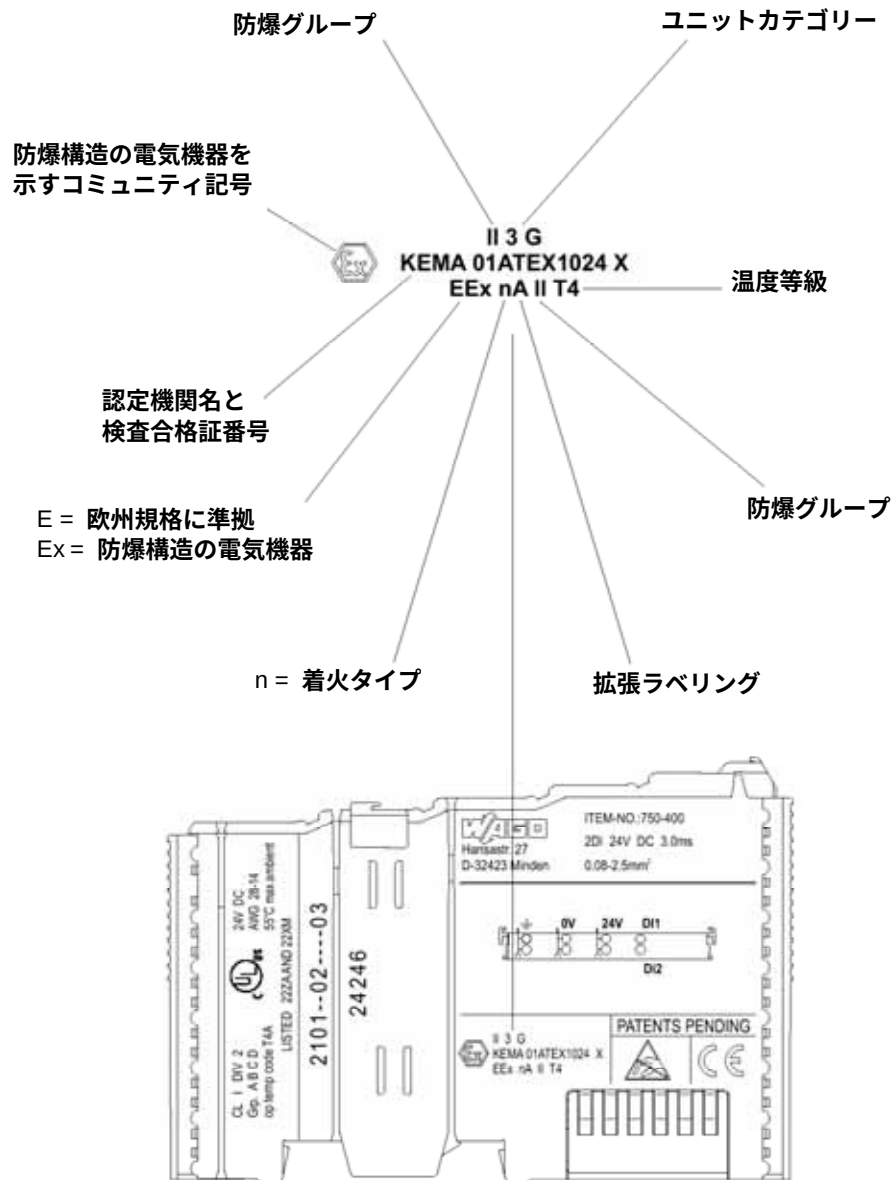


図 6-1：バスモジュール側面のラベリング例
(750-400、2 チャンネル式デジタル入力モジュール、24VDC)

g01xx03e

6.5.2 北 米

NEC 500 によるラベリングの例を以下に示します。

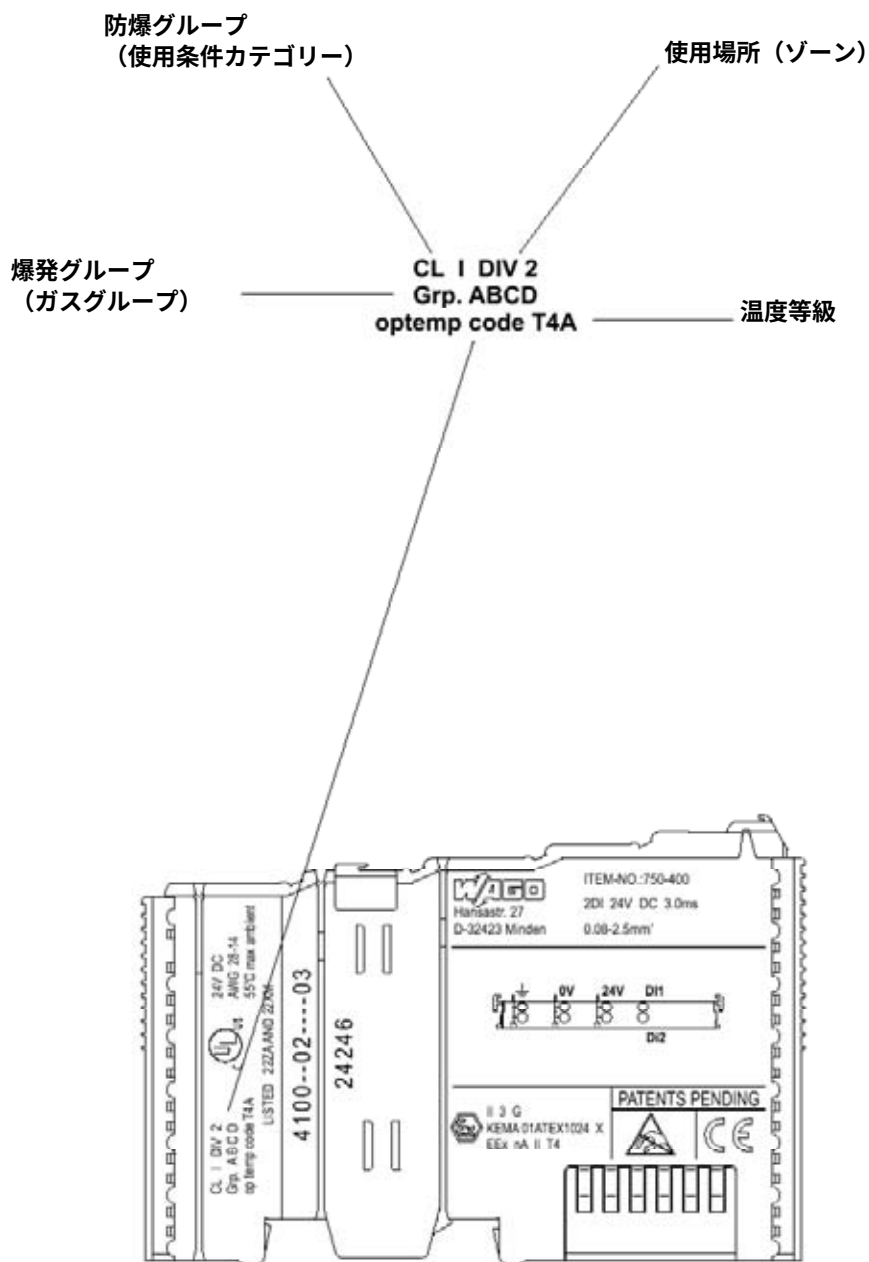


図 6-2 : バスモジュール側面のラベリング例
(750-400、2 チャンネル式デジタル入力モジュール、24VDC)

g01xx04e

6.6 設置規制

ドイツ国内では、爆発性環境での設置に関していくつかの国内規制が適用されます。基本となるのが ElexV であり、これは設置規制 DIN VDE 0165/2.91 によって補強されています。その他の主な VDE 規制を以下に示します。

DIN VDE 0100	発電所における設置で、定格電圧が 1kV 以下
DIN VDE 0101	発電所における設置で、定格電圧が 1kV 超
DIN VDE 0800	情報処理装置を含めて通信設備における設置と運用
DIN VDE 0185	避雷システム

米国とカナダには独自の規制があります。その代表的なものを以下に示します。

NFPA 70	NEC 500危険区域
ANSI/ISA-RP 12.6-1987	推奨される行動規範
C22.1	カナダ電気規程

**危険**

Ex 認定を受けた WAGO-I/O-SYSTEM 750（電氣的運転手段）は、以下の点を満たす必要があります。

- A. フィールドバスに依存しない 750-xxx タイプの I/O システムモジュールは、少なくとも IP54 の侵入保護等級をもつ筐体内に設置する。
ただし、可燃性ダストが存在する場所で使用するときは、上記モジュールを少なくとも IP64 の侵入保護等級をもつ筐体内に設置する。
- B. 電氣的運転手段は、爆発の危険性がある領域（欧州ではグループ II、ゾーン 2。米国ではクラス I、区分 2、グループ A、B、C、D）における使用、または爆発の危険性がない領域における使用にのみ適合する。
- C. 電氣的運転手段については認可されたモジュールのみを使用する。モジュールを交換すると、爆発の危険性がある環境で使用するための適切性が損なわれる可能性がある。
- D. 電氣的運転手段の切断および接続は、必ず電圧供給を遮断した状態、または爆発性雰囲気がないと確認された段階で実施する。電源電圧やヒューズについては指定された値を順守する（ヒューズホルダに記載されたデータを参照してください）。
- E. 本質的に安全な EEx i モジュールをゾーン 0+1 および区分 1 の危険エリアでセンサ／アクチュエータと直接接続して使用する場合は、DC 24V の電源 EEx i モジュールを使用する必要がある。
- F. DIP スイッチとポテンシオメータは、そのエリアに危険性がないとわかっているときにのみ調節する。

**詳細情報**

合格証明が必要な場合はご請求ください。

モジュールの技術情報シートに記載されたデータにも注意してください。



ワゴジャパン株式会社

■本社：〒136-0071 東京都江東区亀戸 1-5-7 日鐵 ND タワー
TEL: (03)5627-2050(代) FAX: (03)5627-2055
ワゴI/OシステムHP : <http://www.wago.co.jp/io/>
